IBM Installation Newsletter

June 16, 1967 Issue No. 67-12

CONTENTS

(G means Green Section)

(continued on next page)

DPD HQ, White Plains, N. Y.

TABLE OF CONTENTS (continued)

Distribution: Branch Offices - DP Management, Salesmen, Systems Engineers, FE Managers. Regions, Districts, Education Centers, Field Systems Centers, Federal Systems Center, FE Area Offices, DPD HQ, FED HQ, WTC.

* Requires Immediate Attention

SELECTED SHORT SUBJECTS

The purpose of Selected Short Subjects is to bring together and highlight concise, factual and timely information which will indicate that action is to be taken by the IBM representative whose accounts are affected.

2848 Physical Planning - Remote Unit

If you are planning to install 2848's remote from the S/360 CPU, i.e. attached to the CPU through a 2701, you should be aware that this control unit has the same temperature requirements, the same maximum relative humidity and maximum wet bulb temperature and the same BTU/hour heat output as a S/360-30 CPU. See "IBM S/360 Installation Manual - Physical Planning" (C22-6820).

S/360 CPU with 2250 and 2840 Meter Operations

The S/360 CPU meter operation article on page 24 of Installation Newsletter issue 66-010 states under item A1:

> "When the CPU meter stops, all meters for Assignable Units and On Line Units stop whether or not the specific actions indicated under their Sales Manual discussions have been taken."

The basic ground rule for graphic unit metering is that the meter will run whenever the buffer is active (i.e. regenerating a display or looping while awaiting an operator action) regardless of CPU meter status.

1800 System Installation of Non-IBM Equipment

The IBM FE Area Installation Planning Engineer should be consulted whenever a customer desires to install non-IBM equipment in, or physically attached to, units of the 1800 system. Reference item 7, page 1 in the 1800 Physical Planning Manual A26-5922-4, or A26-5922-2, -3 with TNL N26-0177.

OS/360 RELEASE 11 CHECKPOINT/RESTART

The following is a correction to Release 11 for Checkpoint/Restart prose, page 12, item 13:

Any number of checkpoints which cause an EOV on a Checkpoint/Restart data set will result in an 806 ABEND.

OS/360 RELEASE 11 - USER LABEL EXITS

The recent Program Announcements Letter (P67-41, 5/15/67) announcing the information pertinent to OS/360 Release 11 also discussed the deferring of Label Exits for User Modifications. This information refers to the use and processing of Volume, Header and Trailer User labels.

OS/360 QTAM - USE OF LOGGING FUNCTION

OS/360 QTAM users currently utilizing the logging function must specify the DCB parameter RECFM=F. Any other specification will result in modification of a portion of QTAM's message prefix. The log function itself will execute properly, however, it could have an unpredictable effect on other functions performed in both message control and message processing. It is planned to lift this limitation in QTAM in a future OS/360 Release during 3Q1967. Actual availability will be as announced with the future Release.

OS/360 - 2314 F.E. DIAGNOSTICS

It is possible for the CE to run micro program diagnostics in the 2314 while the CPU is also running problem programs which use the same 2314 Direct Access Storage Facility. If this is done, the result can be that a control unit busy signal is sent to the CPU because of an attempt to start the data transfer seek while the F.E. diagnostics are also using the control unit.

This condition will cause the problem program to go into WAIT state and shortly the CPU also, since IOS cannot recover from such a condition.

The F. E. Area Office can contact FE Technical Operations, Poughkeepsie, New York for information on this situation. Branch Office communication is through the F. E. Area Office.

OS/360 PL/I (F) PERFORMANCE IMPROVEMENTS

Version 3 of PL/I (F) compiler was recently announced by Program Announcement P67-45, dated May 22, 1967 for October 31, 1967 availability. This release has been devoted primarily exclusively to improving the object time of PL/I programs. See P67-45 for additional information.

1. Many operations which were previously performed by library routines are now executed in-line. These include the more common datatype conversions and certain string operations. The SUBSTR built-in function is now always executed by in-line code. In order to make this possible, the operation was re-defined and the extensive checking inherent in the previous definition was removed. Therefore, care must be exercised that the function does not address a sub-string beyond the source string.

   e.g.
   ```
   DCL ADDRESS CHAR (20),
       CITY CHAR (5);
       CITY = SUBSTR (ADDRESS, X, 5);
   ```

   If X is greater than 15, the variable CITY will contain useless data.

   The savings resulting from this are dramatic: consider the following case:

   ```
   DCL (A, B, C, D)   CHAR (10),
       RESULT         CHAR (20);
   RESULT = SUBSTR (A, 3, 7) || SUBSTR
      (D, 2, 5) ||SUBSTR (B, 2, 3)
                ||SUBSTR (C, 1, 5);
   ```

   The above statement previously compiled into 412 bytes of object code, including 14 library calls. The in-line code produced for this statement as shown below occupies 58 bytes, and, as the concatenations are also done in-line, all 14 library calls are eliminated.

   ```
   LA      8, A+2
   LA      7, D+1
   MVC     WKS(7), 0(8)
   MVC     WKS+7(5), 0(7)
   LA      8, B+1
   MVC     WKS+12(12), WKS
   MVC     WKS+24(3), 0(8)
   LA      8, C
   MVC     WKS+27(15), WKS+12
   MVC     WKS+42(5), 0(8)
   MVC     RESULT(20), WKS+27
   ```

   The object-time performance in this case was improved by a factor of over 15. The compilation time was unchanged.

2. Tests indicate that on some programs the gains from Version 2 to Version 3 are as high as 5:1 overall. In most cases, however, a 20 percent improvement can be expected for commercial programs.

3. Although prolog-epilog overheads have been reduced, in some cases gains may be obtained by re-structuring programs. The dictionary size is effectively doubled for many types of programs by re-use of temporaries and work areas acquired by the compiler. This means that much larger programs may be compiled, thus avoiding excessive segmentation.

   When segmentation was essential, Version 2 presented problems by its use of the Linkage Editor control card INCLUDE. This was especially so when OVERLAY techniques were employed. Version 3 does not generate INCLUDE cards and this problem, therefore, will not arise. A better method of selective routine inclusion is used.

4. The EVENT option on an I/O statement now permits overlap of access time with processing, without buffering. Note that when an I/O statement has the EVENT option, the event specified must be the subject of a WAIT before any I/O conditions (e.g. KEY, RECORD, etc.) are raised.

   e.g.
   ```
   R:READ FILE (PAYROLL) INTO
   (INAREA) EVENT (GETNEXT);
    ||||
   process
    ||||
   WAIT (GETNEXT); /* AT THIS POINT
   CONDITIONS WILL BE RAISED*/
    ||||
   GO TO R;
   ```

   If an event is not WAITed on, then conditions will be lost.

5. Following are the results of timing tests conducted on commercial-type programs. These were inter-related programs and the objective was one of overall improvement of the five programs.

   Although these tests were run under careful supervision, slight errors may be present due, for example, to I/O errors which the system recovers after several attempts. This explains why, in one case, Version 3 of F Compiler is shown as 1.4 seconds slower than Version 2 (22.1 versus 20.7).

   a. Execution time in seconds of poorly coded PL/I Programs for Commercial Applications (time in seconds and OPT=1).

	Pgm 1	Pgm 2	Pgm 3	Pgm 4	Pgm 5
F Version 2	296.5	70.0	352.8	226.7	323.7
F Version 3	102.4	28.9	104.1	49.6	95.0

   b. Execution time of above programs after certain source language improvements: (Time in seconds and OPT = 1).

V2	219.1	151.0	17.9	21.5	201.3
V3	55.4	26.3	17.7	21.6	45.2

   c. Execution time after 'experts' improved the source code to its best. (Time in seconds and OPT = 1.)

V2	24.4	23.0	19.8	20.7	105.7
V3	22.2	22.8	19.9	22.1	28.0

   Poorly coded Version 2 total time - 1270.7 seconds.

   Expertly coded Version 3 total time - 115.0 seconds.

6. With the addition also of a batch-compiling facility, the F Compiler now provides the power of the PL/I language efficiency at the 44K level of OS/360.

7. One other change in Version 3 is of great importance to all users. The PL/I attribute DEFINED is very restrictive to preserve the language's machine independence. In COBOL or FORTRAN, a programmer could use REDEFINES or EQUIVALENCE and, providing he was aware of and obeyed the particular computer's boundary rules, he was able to overlay any datatype with any other. With Version 3, F Compiler will continue to diagnose any illegal use of DEFINED as an error, but code will be compiled and, providing OS/360 boundary alignment is satisfied, the object program will execute correctly. Commercial users employing DEFINED structures as record variables should also familiarize themselves with the mapping of PL/I structures documented in the OS/360 PL/I (F) Programmer's Guide (C28-6594-1).

It is expected that many more users will begin to use PL/I now that the object performance has been improved. However, novices in PL/I should be referred to the Coding Techniques section of the Programmer's Guide (C28-6594-1) since, with a language as large as PL/I, there are often many ways to solve a problem but not all are equally efficient.

OS/360 LINKAGE EDITORS

Release 11 of OS/360 contains both Linkage Editor E, with options for 15K, 18K or 44K operation, and Linkage Editor F with options for 44K or 88K operations. Linkage Editor E is considered to be a most reliable program, and its use is definitely recommended. Linkage Editor E (44K) has several improvements over the 18K and 15K options. The major ones are the ability to accept as input the text blocking of up to 3072 bytes, and the grouped control sections produced by Linkage Editor F.

A comparison of Linkage Editors E (44K) and F (88K) shows the following differences:

	E(44K)	F(88K)
1. Maximum Text Block Size	3072 bytes	18,432 bytes
2. Ability to group CSECTS in output text records	no	yes
3. Maximum segments in structured overlay	63	255
4. Maximum capacities for ESD, intermediate text records, and RLD records	Variable, depends on amount of available core storage	fixed

The structured overlay TTR list, CALLS list, and ENTRY Table sizes also vary somewhat between Linkage Editor E (44K) and Linkage Editor F. (Note: Details are given in Technical Newsletter N28-2224, which amends the OS/360 Linkage Editor SRL C28-6538.)

When the PCI fetch option is selected for the OS/360 Control Programs, there is little improvement in the speed of program fetch due to either grouped CSECTS or text blocking of over 3072 bytes.

The speed of linkage editing when comparing E (44K) and F (88K) depends upon the object modules being edited. Some test runs showed E (44K) to be faster than F (88K) for Assembler, slower for FORTRAN H, and about equal for COBOL F object module editing.

The general conclusion, to the degree that a general conclusion is possible, is to recommend the use of E (44K) except where the number of overlay segments must be greater than 63. A report from you of specific customers who require use of the F Linkage Editor is requested, and may be sent to: Manager, Programming Systems-OS, DPD HQ, White Plains.

Users should be encouraged to convert to Linkage Editor (44K). To do this, long text records (from Linkage Editor F) must be converted to short records acceptable to Linkage Editor E (44K). Following is the general procedure which would be used to convert:

1. For the user who utilizes the IBM supplied DLIB01 and DLIB02 (SYSRS2, SYSRS3) and has private libraries of 1K or 3K byte records, the normal generation procedures will suffice.

2. For the user who has records longer than 3K bytes on a 2301, 2302, 2303 or a 2314, Linkage Editor F 44/88 in his existing system must be used to link edit with the DC (Downward Compatible) option. The output may be written on any direct access device. Then, using Editor E 44, he must link edit the intermediate output to his new system.

   Exhibit 1 shows the conversion procedure.

OS/360 RELEASE 11 - USE OF NEW UTILITY

Users who desire the new update utility (IEBUPDTE) must linkedit it separately because SYSGEN generates the utility (IEBUPDAT) into SYS1.LINKLIB. See the OS/360 Release 11 prose for JCL.

DOS/360 DASD FILE PROTECTION

Direct Access storage device file protection (DASDFP) may be specified in the FOPT system generation macro for DOS/360. DASDFP operates in the following manner:

1. Job Control or the Foreground Initiator reads the VOL, DLAB, and XTENT control cards from the input job stream. This information is re-formatted somewhat, and written on the label storage area of SYSRES as one or more disk records. Each of these disk records is written with (preceded by) a disk key. This disk key is created from the "filename" (DTF name, really) entry in the VOL card.

2. When the problem program OPENs the data set (DTF), the OPEN transient routine extracts an 8-byte alphameric constant from location 22-29 of the generated DTF table. This constant is called the "filename", and is an exact replica of the DTF name.

3. The OPEN transient routine searches the label storage area of SYSRES looking for a disk key equal to this DTF "filename" constant. When it finds a match, it reads the data portion of this job control disk record into memory.

4. The extent information is extracted from this job control disk record, and is stored in the JIB table. A 2311 extent requires one JIB entry; a 2321 extent requires two JIB entries.

   Each XTENT card named the SYSnnn unit that contained the extent. The $$BOFLPT OPEN transient, therefore, stores each extent into a JIB entry that is related or linked to a specific SYSnnn or logical unit block (LUB).

OS/ 360 LINKAGE EDITORS

LKED F44/88>3K Records to LKED E44≤3K Records

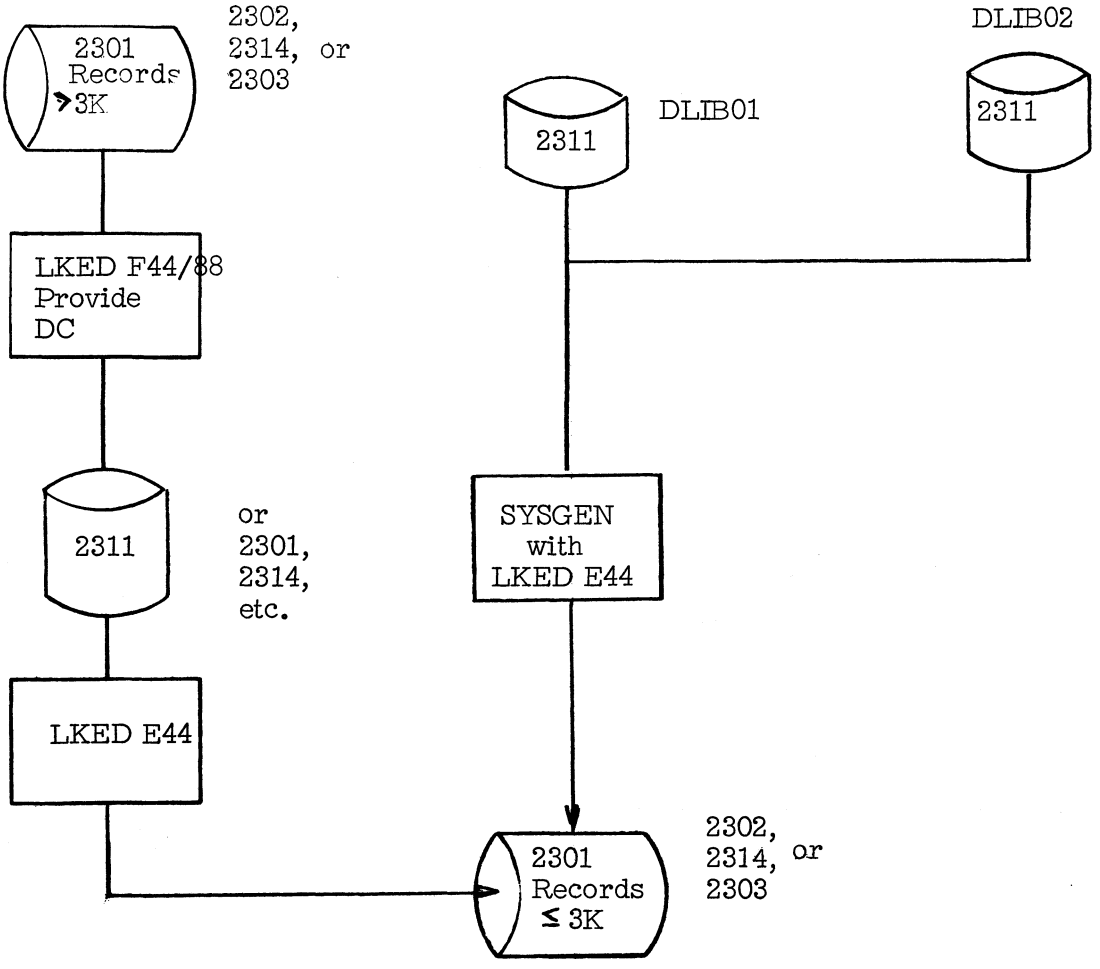


Exhibit 1

5. When the problem program requests an I/O operation on disk, the supervisor extracts the SYSnnn and the SEEK channel command word (CCW) from the channel control block (CCB). The current disk SEEK address is compared against the extent limits stored in the JIB table for this specific SYSnnn. If the disk SEEK address falls outside the range of extents, the job is cancelled.

   For a 2311 extent, the upper and lower cylinder numbers comprise the protection boundaries for the named SYSnnn. A 2321 is only protected on strip boundaries.

NOTE: This is a programmed check to determine if the problem program is trying to access data outside the "allowed" cylinder or strip limits.

6. Next, the supervisor builds and executes a small channel program containing three commands: a SEEK that is identical to the problem program SEEK; a Set File Mask that prevents any other "long" SEEKs from being executed; and a TIC that transfers control to the command following the problem program's SEEK.

   NOTE: This is a "hardware" check to prevent the problem program's channel from moving the disk R/W head outside the cylinder or strip limits that were validated by the "programmed" check.

DASDFP provides file protection on the basis of Symbolic Unit. It does not provide protection by access method or data set or DTF. However, if each disk DTF in a problem program is assigned different SYSnnn units, then file protection can be "made" to prevent one DTF from accessing the data belonging to another DTF.

Also, DASDFP does not prevent data set contention between partitions. Thus, more than one partition may access the same data set at the same time (and both partitions may attempt to update the same disk record simultaneously).

When using DASDFP, follow these guidelines:

1. For complete protection, data sets should begin and end on cylinder/strip boundaries.

2. File protection is ensured only if the disk labels involved are unexpired.

3. In any one program, each DTF should use a different SYSnnn, even if the data sets reside on the same physical volume.

DOS/360 INSTALLING AND TESTING TELE-PROCESSING

A member of an IBM Field Systems Center has been assisting field System Engineers in the installation and testing of Tele-processing systems operating under the support of DOS/BTAM. This article is the result of this experience. It describes common errors which occurred in installing/testing and how to avoid them. In addition, some suggestions for easing the installing/testing phase are included.

The following are some common DOS-BTAM installing and testing errors.

1. Incorrect SAD commands issued for 2702 lines. Each 2702 communication line is wired to accept a specific SAD command. The command (SAD0, SAD1, SAD2, SAD3) specifies a given terminal control and bit rate. The wiring of the SAD command to be used for each line may be done at the plant or by the Customer Engineer at installation.

   SAD commands are issued in accordance with the MODE setting specified in DVCGEN statements or ASSGN statements. Prior to installation, a meeting should be held with the Customer Engineer to determine the specific SAD commands that will be utilized for each line on your 2702.

2. Incorrect device addresses specified for 2702 lines. As a general rule, line 0 is reserved for Customer Engineering diagnostic tests (wrap tests) and is not physically wired for user operations unless the full capacity (15 or 31 lines) of the 2702 has been reached. Therefore, except for a full capacity 2702, the first customer useable line will be line 1. Line 0, however, will take up the first channel device address of the control unit.

   As in item 1 (above), you should meet with the Customer Engineer to determine channel addresses for each useable line prior to systems generation and testing.

Exhibit 1 is an example covering items 1 and 2. Assume a 2702 with 3-1050 lines and 2-1030 lines.

A helpful planning and testing aid is to lay out a chart such as is shown in Exhibit 2. This chart would also be an aid to Customer Engineers when de-bugging hardware. One chart should be filled out per communication line.

3. Incorrect Dataset interface and incorrect Dataset options. Dataset interface circuitry is provided in the Transmission Control Unit on a per line basis. It is built to match a specific Dataset and is determined by the information the sales representatives write up on the 2701 or 2702 Specification Sheet. The Specification Sheet must be submitted prior to PSC time. This information must be accurate to assure an on-time installation.

   Dataset options are to the Common Carrier what special features and "specifies" are to the IBM Corporation. They define just how the Dataset is to operate on the circuit and with the IBM equipment. The IBM representatives responsibility to the customer regarding the specific Datasets and options to be ordered from the Common Carrier is contained on page GI2 of the Sales Manual. Common Carrier information is detailed in the IBM Data Communications Handbook. Additional information is available from your District FSC SD&I Department. Dataset availability and requirements should be determined as early as possible. Some Datasets are in short supply at the present time and a lead time of five months exists in some instances.

4. Excessive 'lost data' errors when reading from a terminal. Lost data occurs when the count of characters received exceeds the count set up for the read operations. Many users are forgetting to include the EOA character (circle D) and the EOB character (circle B) in the read count. Lost data is also likely to occur when reading from online keyboards because some operators will tend to include extra spaces. The human element should be taken into consideration, and a large enough input area for some extra characters allowed.

5. Excessive "data checks" when writing to a terminal. Check the EBCDIC to terminal code translation table for codes not acceptable for that terminal type. For example, many users are placing a X'00' in the table wherever a non-existent terminal code logically appears. Thus, a EBCDIC X'00' will be translated to terminal code X'00'. This will cause an error if transmitted to a 1050. A better approach is to use the hex representation for some special character which will be readily identifiable at the terminal. An asterisk might be a suitable choice.

6. Extraneous coding in the I/O area. This will generally show up in a storage dump taken at the start of program operation and is caused by setting up I/O area with DS statements. If it causes program problems, the "ACTION CLEAR" option should be used when link editing the program.

7. Failing to transfer control to a lower priority partition after executing T/P WAIT. The WAIT macro scans the DECB's in the list for completion codes. Every time a completion code is found in the scan, one is subtracted from the users WAIT count (normally 1). When the count is reduced to zero, the WAIT is satisfied and control is returned to the users T/P program. If the count is not reduced to zero, control is passed to the supervisor for task selection.

   If you find the program failing to transfer control to a lower priority program, take a storage dump and check the DECB's. The user has probably stopped operations on a line and failed to clear the DECB completion code. This condition will most likely occur on dial-up lines where one may terminate a call on one line and not start another.

8. T/P program in continuous WAIT state. This can occur under several conditions.

   a. Executing a WAIT with no I/O operation started - this is a program logic problem. It will be necessary to cancel the job and correct the logic because there is no way to regain control.

   b. Executing WAIT for I/O having an indeterminate completion time. This will occur only when all lines are performing READ

initial operations and only if all the lines are supporting any of the following:

1) Audio Response
2) 2260 Local
3) Polling with wrap lists
4) 2740 Basic
5) Auto-answer

The only way to regain control is for some terminal to respond positively by sending in a message, thus satisfying the WAIT. A local terminal is virtually a must in systems of this type. Operator Control (OC) and/or Interval Timer (IT) STXIT routines will not circumvent this problem because they do not gain control until the partition gains control - in other words, until the partitions WAIT condition has been satisfied.

The following are installing and testing suggestions.

1. Make sure the system will pass the Customer Engineering diagnostic check out programs before you begin initial testing. These programs prove invaluable when machine malfunction is suspected.

2. Include the terminal test facilities in your BTAM logic and run them before each test session.

3. Message control (line control) and message processing should initially be tested separately.

   Message processing routines can be tested by simulating message input via a card reader or magnetic tape file. Output messages can be simulated by writing to the printer.

   Message control testing should start with single thread testing. Typically, this would involve reading a message from one terminal at a time, identifying the terminal from which the message was received and printing the message and terminal identification out on the system printer.

   The next test might involve receiving a message from one terminal at a time, identifying the terminal, and then turning the line around and transmitting the message back to the terminal.

   Further testing of message control would involve multiple line operations, operator control, changing of polling lists, etc. The point is to take one step at a time, prove out the step before going to the next step and adding to the complexity of the system.

   Only when the separate packages (message control and message processing) have been satisfactorily tested independently should they be combined. At this point, the user should again start with single thread testing - one terminal and one transaction at a time.

4. Liberal use of the PDUMP macro will enhance T/P program testing. Many programming errors are found in the execution of line I/O macros (READ, WRITE, CONTROL, etc.) and in the handling of I/O areas. In most systems the multiplicity of lines and terminals plus the random arrival of messages and completion of events makes it very difficult to pin down these programming errors. What is required is a trace tool. PDUMP is a very effective trace tool for this type of situation. Users should insert PDUMP macros before and after execution of any line I/O macro. The dump should be set up to print out the DECB's, I/O areas, and DTF's.

   Users should be aware that PDUMP uses registers 0 and 1 and consequently destroys their contents. These registers, therefore, should either not be used in the program, or their contents saved before execution of PDUMP and restored after execution.

DOS/360 BTAM LOCAL 2260 TERMINALS

In a DOS/360 system supporting local 2260 terminals under BTAM, it is usually desirable to establish a technique for operating with less than the total number of terminals actually attached to the system. This requirement might exist in one or more of the following situations:

1. During initial testing of the program when only one or two terminals have been installed.

2. At any time that a terminal has been taken offline for maintenance.

3. When operators are not available to man all terminals.

DOS/360 INSTALLING AND TESTING TELE-PROCESSING

Channel Address	2702 Line	Terminal Control
010	0	not assigned
011	1	1050-SAD0
012	2	1050-SAD0
013	3	1050-SAD0
014	4	1030-SAD1
015	5	1030-SAD1

DVCGEN statements for this situation would be:

```
DVCGEN    CHUN=X'011',DVCTYP=2702,MODE=X'00'
DVCGEN    CHUN=X'012',DVCTYP=2702,MODE=X'00'
DVCGEN    CHUN=X'013',DVCTYP=2702,MODE=X'00'
DVCGEN    CHUN=X'014',DVCTYP=2702,MODE=X'01'
DVCGEN    CHUN=X'015',DVCTYP=2702,MODE=X'01'
```

ASSGN statements might be:

```
ASSGN  SYS020,X'011'
ASSGN  SYS021,X'012'
ASSGN  SYS022,X'013'
ASSGN  SYS023,X'014'
ASSGN  SYS024,X'015'
```

The DTFBT macros in the users program might be:

```
Group 1    DTFBT    LINELST=(020,021,022),
                    CU=2702,
                    DEVICE=1050
Group 2    DTFBT    LINELST=(023,024),
                    CU=2702,
                    DEVICE=1030
```

Exhibit 1

DOS/360 INSTALLING AND TESTING TELE-PROCESSING

HARDWARE

2702 Line # _____ Channel Address _____ Terminal Type ____ SAD ___

COMMUNICATION

Line Adapter or Dataset Type _____ Telephone or leased line# _____

PROGRAM

Sys # ____ DTFBT name _____ Relative Line _____ DECB name _____

TERMINALS

Hardware Address	Physical Location
_____	_____
_____	_____
_____	_____
_____	_____
_____	_____
_____	_____

Exhibit 2

4. During off hours when only a few terminals are in use.

Do not use the DOS ASSGN IGN statement as a means of eliminating the terminals not in use. Since ASSGN IGN forces completion of a READ command when an out of service terminal is tested, the use of this statement will prevent BTAM from testing for ENTER from some active terminals. Any active terminal which appears in the scanning sequence after a terminal for which ASSGN IGN has been specified will be ignored by BTAM.

The recommended alternate to ASSGN IGN is the use of the skip bit in the line control block for terminals not in use. When this bit is on, it prevents polling of the associated terminal. The skip bit can be turned on by the BTAM program, usually through use of the CHGNTRY macro instruction. The use of the skip bit means that terminal configuration on modifications must be made from inside the program rather than externally by the operator as would be the case with ASSGN IGN. This implies that the programmer will code into his BTAM program the necessary routine to accept configuration information (probably from either the console or the card reader) and to set the appropriate skip bits.

DOS/360 BTAM 2702/2703 AUTOPOLL SUPPORT

BTAM support for the autopoll feature will allow the user to specify either an open list or a wrap list for his terminals. The use of an open list allows the user to poll each terminal in the list once and then to regain control in his program from the autopoll hardware. In most instances an open list will be used in conjunction with the timer. The program will utilize the autopoll hardware to poll each terminal once and then will set an interval on the timer during which polling will be suspended. BTAM channel programming with a wrap list will allow polling to continue indefinitely without any program intervention. The polling list may vary in size from a single entry to many entries. Even with only one entry in the list, the wrap list channel program will continuously poll that terminal without reference to the CPU program. When using a wrap list, polling need not begin with the first entry in the list. No matter where in the list the polling sequence is started, the channel programming will always wrap back from the end to the beginning of the list. The use of the 2702 Autopoll special feature, when BTAM support becomes available, is strongly recommended. By eliminating the CPU time required by BTAM to re-initiate polling after a negative response has been received, this feature provides a significant saving in CPU time for all but the smallest T/P networks.

DOS/360 ISAM ON 2321 DATA CELL DRIVE

The following article concerns ISAM data sets on the 2321 under DOS/360 and other information about the 2321.

1. An ISAM data set must be CLOSEd if records are being LOADed, or if ADDitions are being entered. Occasionally, it is not possible to CLOSE an ISAM data set, because some I/O errors cause the supervisor to cancel the job without returning control to the user.

   In a LOAD operation, the CLOSE writes a completed F2 Disk Label. If the ISAM data set is not CLOSEd, it is unusable, since the unwritten F2 Disk Label would have carried the addresses of the master and cylinder indexes, the number of the cylinder overflow tracks, a pointer to the independent overflow area, etc.

   In an ADDition operation, the CLOSE writes an updated F2 Disk Label with the current address of the next available location in the independent overflow area. If the data set was not CLOSEd, this pointer would still have its old value; and future ADDitions would overlay the ADDitions just inserted into the independent overflow area.

   Also, the job might be cancelled while an ADDition (either to a cylinder or to an independent overflow area) was being handled by ISAM. An ADDition often requires several disk reads and writes to completely update the "chain" of disk address pointers. If the overflow record chain is not completed, some records are usually lost. The incorrect overflow chain produces a data set that cannot be processed sequentially.

   Thus, the user is faced with the unhappy prospect of reconstructing the ISAM data set. While the time to reload or reconstruct an ISAM data set on a 2311 is mostly a nuis-

ance, the time to reconstruct an ISAM data set on a 2321 is very significant.

The following are a few possible techniques to lessen the consequences of an unpredictable supervisor cancel:

a. Modify the CCB in each problem program DTF to insure that the bit which instructs the supervisor to return all unrecoverable I/O errors to the user is on. This will provide the ability (in most cases) to CLOSE all files before cancelling the job on I/O errors.

b. Split the logical file into smaller physical data sets, to reduce the amount of reconstruction effort in the event of an unCLOSEd condition.

c. Take frequent checkpoints and maintain an adequate audit trail of all transactions, to ease the task of reconstruction.

d. Do not create an ISAM data set on a 2321. Instead, use DAM to create and maintain the 2321 data. Use the 2321 physical addresses to create a separate 2311 ISAM data set, that will actually be an index to the master data on the 2321. This (DISAM) organization is further discussed in TIE paper Z77-6247.

An unCLOSEd DAM data set can be reconstructed since the Disk Labels would not be altered by the CLOSE, and there is no uncompleted disk "chain." If the 2311 ISAM data set (really an index to the 2321 DAM data set) was unCLOSEd, the user would still have the problem of its reconstruction. However, this isn't nearly as great a task as that of reconstructing an entire 2321 ISAM data set.

2. It is very strongly recommended that the user VERIFY all records written on the 2321. A flexible media, like magnetic tape or strips, is more susceptible to write errors than is a hard (2311, 2314) surface. A write VERIFY could eliminate a difficult reconstruction later.

3. Be prepared for alternate track assignment. Plan how you are going to reconstruct the data from the bad track. To insure a smooth installation, practice this procedure with the customer personnel.

4. Two Type III program abstracts on ISAM were published in Installation Newsletter 67-05, page 24:

a. OS/360 Generalized ISAM Utility Program

b. ISA 360 - Index Sequential Analyzer on S/360

5. Publications

a. Component Descriptions (2841 et al)	A26-5988-3
b. Student Text-Introduction to DASD & Organization Methods	C20-1649-1
c. Data Cell Handling Guide	A26-3622-1
d. 2321 F. E. Theory of Operations	227-5895-1
e. Capacity & Transmission Time Reference Cards	X20-1704
f. DASD Workshop - Instructors Guide	Z23-1522
g. File Organization Techniques for DASD	Z20-0328-2

6. Installation Newsletter Items

Issue	Title	Page
66-021	Calculating DASD Space Requirements for Indexed Sequential	19
66-022	DOS/360 2321/2311 Considerations	
	1. Logical IOCS (Indexed Sequential)	3
	2. Initialize 2321 Utility (latest level)	4
	3. Physical IOCS	4-5
66-023	2321 Error Recovery Procedures	25
	2321 Prevention of Data Loss & Data Cell Damage (internal reset restrictions)	26
67-004	DOS/360 ISFMS Pointers	23-24
67-005	OS/360 SYSGEN Specifications for 2321	2
	OS/360 Use of 2870 Selector Subchannels w/2321 not Supported	2

7. TIE Papers (see abstracts in KWIC Index to TIE)

a. "One Approach to Variable Length Records on S/360 DASD." Z77-6260
b. "A Thruput Analysis of the IBM 2321." Z77-4952
c. "A File Organization Technique for the IBM 2321." Z77-4754
d. "A Master File Organization for all Insurance Applications Utilizing the IBM 2321." Z77-6002
e. "When to Reorganize a Sequential Data File." Z77-5031
f. "The Effect of Activity Distribution on the Throughput of Sequential Batch Processing Systems." Z77-6005
g. "Organization of a 2321 with Extension on a 2311 Disk Drive." Z77-6075

8. Miscellaneous

Branch Office Managers Letter 3/12/65 contains some performance parameters and some system design considerations for the 2321.

DOS/360 QTAM ERROR MESSAGES

Most QTAM ERRMSG macros define error messages which are to be sent back to the terminal which was the source of the error. To aid in getting this message back promptly, QTAM makes an error message a priority message. Message priority for QTAM messages in not effective once a message has been committed to a message queue on disk. In order to assure prompt transmission of error messages back to the source, the user must make sure that he has an open line to the source while the error message priority is still in effect. He can do this by specifying send over receive priority in his DTF or DCB for the line group. If he specifies receive over send or equal priority, he in effect allows polling of other terminals on the line to continue after receipt of an error message. This will usually result in a delay in delivering the error message. This message will have to wait for receipt of any additional incoming messages on the line. It will also have to wait for dispatch of any messages ahead of it on the disk queue once its priority status has been lost. Please consider these facts before specifying anything other than send over receive priority for a QTAM line group.

DOS/360 FORTRAN TAPE LABEL CHECKING

Because of FORTRAN language considerations, its IOCS opens all tape files as work files. It is always assumed that files may be written during a run. The OPEN routines (called implicitly by FORTRAN code) thus perform the following operations for all files even though they appear as input only or output only to the user:

1. Read the first record.

2. If it is a tape mark then position past it and process the file.

3. If it is neither a tape mark nor a label then write a tape mark and continue processing. (Thus, if unlabeled, the system will write a tape mark, even if operating with an input-only file. No diagnostic is given, except if there is no ring in the tape when trying to write the tape mark.)

4. If the first record is a label of the form HDRIb...b then continue processing.

5. If the record is an expired label then rewrite it as HDRIb...b. and continue. If it is unexpired, then stop and allow the operator to bypass. (Note that if expired, or unexpired and bypassed, the system will attempt to write a blank label. If this is an input tape, the rewrite of the label could ruin the first data record.)

The system bypasses system TLAB cards. Our suggestion has been to use unlabeled tapes, with a tape mark in front of the first data record. A generalized flow chart of the operation is shown in Exhibit 1.

DOS/BOS/360 REDUCING CIL DISK SPACE REQUIREMENTS

Significant space may be saved in the CIL (Core Image Libraries) of both DOS/360 and BOS/360 when cataloging Assembler Language programs by the use of the following technique. Users should be aware that the technique has not been submitted to any formal IBM test.

The Linkage Editor builds core load disk records as it processes assembler output

DOS/360 FORTRAN TAPE LABEL CHECKING

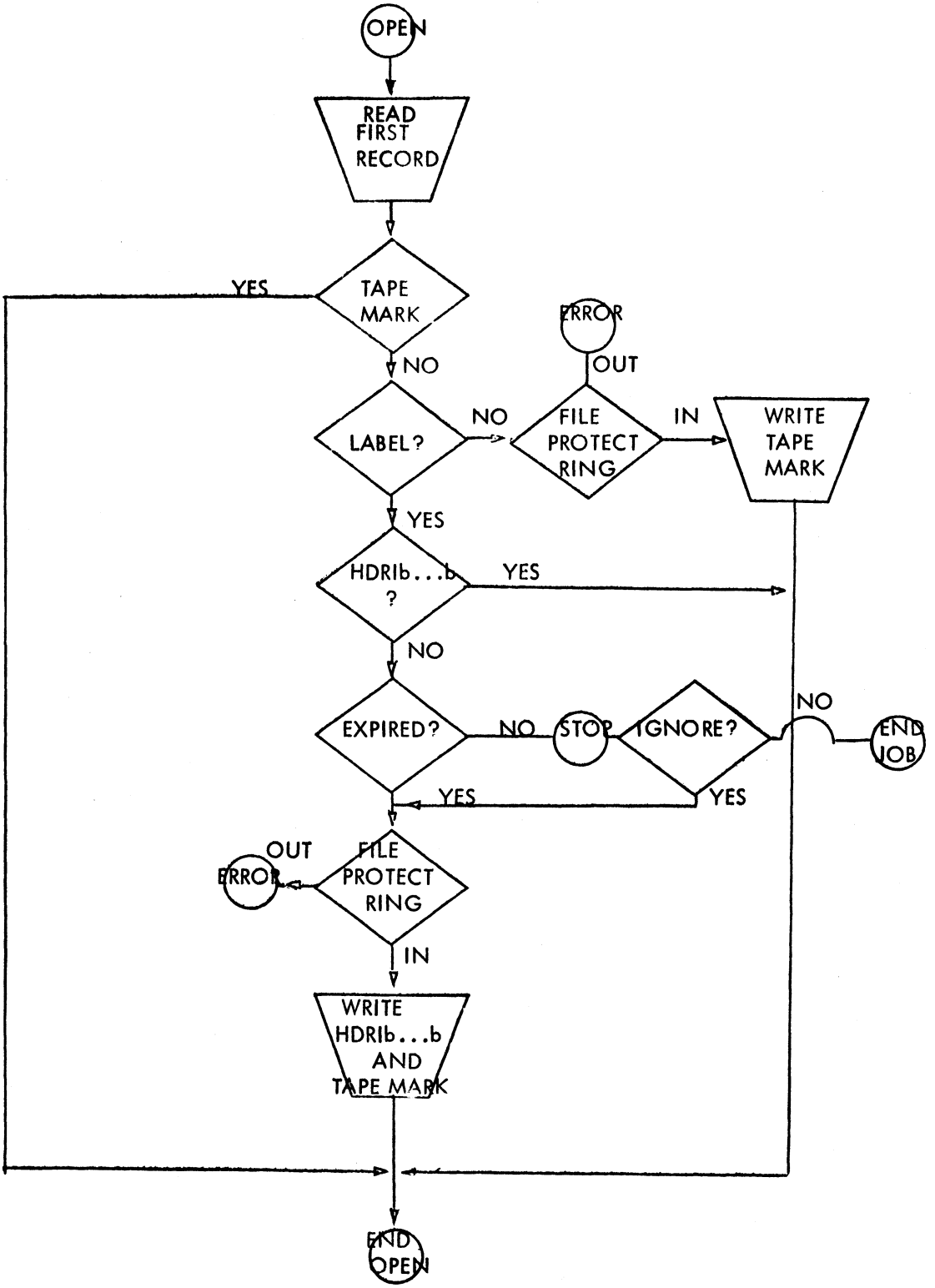


Exhibit 1

(ESD, TXT, RLD, REP, END). The amount of actual disk space required depends upon the highest value reached by the assembler location counter during the assembly. While an object module contains no TXT cards for areas defined by DS statements, Core Image Library space is required for DS defined areas. This means that I/O areas, which will contain valid data only after program execution begins, still occupy valuable space in the Core Image Library and must be read in from disk by the Systems Loader.

To eliminate this usage of Core Image Library space, the user may define his I/O areas by using EQU statements. EQU's do not increment the location counter. The EQU statements must come after a LTORG statement and just prior to the END card.

The following example illustrates the required coding, assuming user constants follow user coding.

```
LASTDC      DC      C'LAST CONSTANT IN
                    USER AREA'
            LTORG
IOAREA1     EQU     *       LENGTH OF 740
                            BYTES
IOAREA2     EQU     IOAREA1+740
TAPEOUT1    EQU     IOAREA2+740  LENGTH 500
TAPEOUT2    EQU     TAPEOUT1+500
DISKIN1     EQU     TAPEOUT2+500  LENGTH
                    3620
DISKIN2     EQU     DISKIN1+3620
LASTCORE    EQU     DISKIN2+3620  LAST
                    LOCATION USED
            END     START
```

The label, IOAREA1, will have the highest value of the location counter and this value will be the limit of space needed in Core Image Library.

This technique, as described, requires no further consideration for BOS/360 users. The AUTOLINK function of DOS/360, however, will present problems at linkage editor time. Relocatable object modules automatically included by AUTOLINK will be assigned core in the areas reserved by the EQU statements. To eliminate the conflict over core, all modules normally autolinked must be explicitly included through the use of INCLUDE cards preceding the first ESD card of the object module.

Determining which logic modules, etc. must be included through INCLUDE cards may be done either by examining the ESID listing of the assembler output for EXTRN names or by actually running an // OPTION LINK job step and seeing from the MAP what modules were autolinked. The INCLUDE cards may be automatically punched with the object module by using the PUNCH or REPRO macro facility of the assembler language.

This technique will greatly reduce the disk space required for DOS/BOS/360 Core Image Library phases. Implementation is relatively simple.

BOS/360 to DOS/360 CONVERSION CONSIDERATIONS

There are several differences between BOS/360 and DOS/360, which must be considered in any conversion activity. Each one by itself is a simple matter, but some problems should be anticipated in an installation with many programs. A recent user conversion effort produced the following list of inconsistencies that must be resolved.

1. Communication region format is different.
   a. No Julian date.
   b. Date is edited in DOS MM/DD/YY which is MM DD YY Julian in BOS
   c. Storage protection restricts MVCOM usage.
2. BOS allows fill usage of registers 12 and 13 if specified at System Generation. Register 13 is required for the CALL and RETURN macros in DOS.
3. Maskable program check interrupts which were enabled by BOS are all masked off in DOS.
4. There is no MSG macro in DOS.
5. The DTFBG and DTFEN cards are not totally compatible in DOS.
6. In assemblies under DOS, character self-defining constants require a C 'X' format, while BOS accepts a 'X' format.
7. Y type Adcons will not work if loaded beyond 65K.

8. There is no XFR macro in DOS.

9. Although DTFSR will assemble under DOS, it should be recoded for assembly speed and potential core savings.

10. The console DTF must be eliminated from OPEN and CLOSE statements.

11. The CHKPT macro can not be used in the foreground partition.

12. Change in system logical name from SYSOPT to SYSPCH.

13. Changes of the AFTERID and RZERO specifications cause logic changes if the BOS system being converted was System Release 7 or earlier.

14. The CCB macro and expanded format are different.

15. If a CONTROL=YES is not specified for the reader, a SSELECT=2 must be included.

16. Index sequential files must test File name C for errors under DOS, while in BOS the programmer specified labels of various error routines and BOS branched to them when the error occurred.

17. Several macro expansions are different and this causes problems in multiple usage of one macro. Furthermore, due to early problems in BOS, programmers coded the macro expansion themselves and thus they are difficult to find in the conversion.

18. Other format inconsistencies:

    a. No IS operand on the ISAM macros.
    b. Special byte may be required for CCB.
    c. EXIT macro has different format.
    d. STXIT macro must be recoded.

19. DLAB and XTENT cards are expanded.

20. BOS-STR is quite different from the interim release of STRAM.

BOS/BPS/360 STR SUPPORT

A S/360 replacing an off-line tape terminal (i.e. 7702 or 7711) and using the BOS/BPS/360 STR support may result in a decrease in throughput on the Tele-processing line. The Read and Write macros each cause two CCW's to be built. The first of these, in each case, is a Test Sync command. The elimination of this CCW from the Read and Write macros will correct the throughput problem.

S/360 EDUCATION - PL/I - ADDITIONAL REFERENCE MATERIAL

1. "Programming Language/One" by Frank Bates and Mary L. Douglas (C 1967 Prentice-Hall, Inc.) is available in a paperbound edition for $5.95 at book stores.

   The book was reviewed by Mr. P. F. May of the DPD Education Development Department and his opinion follows: The text is easy to read and understand, and contains many illustrative examples and exercises. It is good for a first look at PL/I. Instructors planning to include its use in courses should consider the following points:

   a. It appears to be primarily oriented toward the non-commercial OS/360 programmer.

   b. Some of the solutions to exercises do not explicitly illustrate the problem as stated.

   c. Record-oriented input/output is given a cursory explanation, with a few examples and no exercises.

   d. Much use is made of floating-point arithmetic, including the illustration of commercial problems.

   e. Some examples, including some previously compiled and tested by the authors, do not conform to the current implementation of the PL/I language.

2. Three Student Texts, currently available from the Distribution Center at Mechanicsburg, are of greater value as supplementary student material in PL/I classes:

   a. A Guide to PL/I for FORTRAN Users (C20-1637)

   b. A Guide to PL/I for Commercial Users (C20-1651)

   c. A PL/I Primer (C28-6808)

NEW PUBLICATIONS ANNOUNCED BY PRL'S

The weekly PRL's (Publications Release Letters) are used to insure that all Salesmen and Systems Engineers are aware of new or revised Marketing Publications. Normally, each issue of the Newsletter will contain information from two PRL's, one following the other. The information will be placed in the Newsletter in its original form with no rearrangement of form numbers or titles. It is not intended to replace existing information and distribution sources. You should be certain that you are aware of these sources.

- PRL #20 May 19, 1967 -

N20-1410.24	1410 SRL Newsletter Re: A22-6826-3	Scrap N20-1410.23
Z20-0834-0	KWIC Index to Technical Information Exchange - Accumulative Supplement May 1967 Re: Z20-0305-13	Scrap Z20-0828-0

MARKETING PUBLICATIONS

221-0435	System/360 Scientific Subroutine Package Proposal Insert	REVISED ABSTRACT
221-0438	System/360 General Purpose Simulation System Proposal Insert	REVISED ABSTRACT
221-0439	System/360 Project Control System Proposal Insert	REVISED ABSTRACT
221-0440	1130 Project Control System Proposal Insert	REVISED ABSTRACT
250-0044-0	Systems Science Education Program	NEW
250-0045-0	Design and Analysis of Communication-Based Systems	NEW
250-0046-0	Application Material for Enrollment in Systems Science Education Program	NEW
A22-6877-3	IBM System/360 Component Descriptions and Operating Procedures IBM 1052 Printer-Keyboard Model 7 with IBM 2150 Console	Scrap A22-6877-2
A24-1088-1	Original Equipment Manufacturers Information IBM 523 Gang Summary Punch - IBM 526	Use A24-1088-0 & N24-0224
A24-3312-4	2821 Control Unit	Use A24-3312-3, N24-0396 & N24-0399
A24-3476-0	IBM 1401H System Summary	NEW
A24-3494-0	IBM 1401 Model H Configurator	NEW
A26-5922-4	IBM 1800 Physical Planning	Scrap A26-5922-2, -3, N26-0157, N26-0158, N26-0172 & N26-0177
A27-3007-0	IBM 2780 Data Transmission Terminal--Installation Physical Planning	NEW
C24-5038-1	IBM System/360 DOS/TOS FORTRAN IV Programmer's Guide	Use C24-5038-0, N21-5034, N21-5048 & N21-5053
C27-6909-3	IBM System/360 Operating System Graphic Programming Services for IBM 2250 Display Unit	Scrap C27-6909-2, N27-1264, C27-6921-0, N27-1251, N27-1262 N27-1263
C27-6912-3	IBM S/360 Operating System Graphic Programming Services for IBM 2260 Display Station (Local Attachment)	Scrap C27-6912-2, C27-6925-0 N27-1246 & N27-1252
C28-6596-2	IBM S/360 FORTRAN IV Library Subprograms	Scrap C28-6596-1
C28-6656-0	IBM S/360 Operating System: Checkpoint/Restart	NEW
C33-4000-0	IBM S/360 Operating System: ALGOL Programmer's Guide	NEW
E20-0181-1	Executive Summary - Advanced Data Processing for Retailers	Scrap E20-0181-0
E20-0282-0	Data Processing Instruction at Waterford Township School	NEW
H20-0294-1	System/360 Flowchart (360A-SE-22X) Operator's Manual	Scrap H20-0294-0
H20-0340-0	Structural Engineering System Solver (STRESS) for the IBM 1130, Model 2B (1130-EC-03X) User's Manual	NEW
K20-0211-0	Wisconsin Telephone Company Rate & Route Computer System (RRCS)	NEW
N20-1021-0	TNL Re: H20-0237-1 STRESS Application Description	NEW
N20-1023-0	TNL Re: H20-0329-0 for System/360 Product Structure Retrieval Program - Application Program - Application Description	NEW

- PRL #20 (continued) -

N24-5289-0	TNL to DOS System Control & System Service Program Re: C24-5036-1 with N24-5218	NEW Scrap N24-5208
N24-5301	TNL to IBM S/360 DOS PL/I DASD Macros Re: C24-5059-0	NEW
N28-0210-0	IBM S/360 FORTRAN IV Language Re: C28-6515-4 with N28-2147	NEW
N28-0211	TNL IBM S/360 Operating System: FORTRAN IV (E) Programmer's Guide Re: C28-6603-1	NEW
N28-0212	TNL IBM System/360 Operating System: FORTRAN IV (G) Programmer's Guide Re: C28-6639-0 with N28-2212	NEW
N28-2218	IBM S/360 Operating System: Operator's Guide Re: C28-6540-4 with N28-2209	NEW
N28-2221	IBM S/360 Operating System: System Programmer's Guide Re: C28-6550-2	NEW
N28-2223	IBM S/360 Operating System: Supervisor & Data Management Macro Instruction Re: C28-6647 with N28-2217	NEW
N28-2224	IBM S/360 Operating System: Linkage Editor Re: C28-6538-3, -2, -1 with N28-2129, N28-2130, N28-2201	NEW
N28-2225	IBM S/360 Operating System: Utilities Re: C28-6586-4	NEW
N28-2226	IBM S/360 Operating System: Job Control Language Re: C28-6539-4 with N28-2214	NEW
N28-2228	IBM S/360 Operating System: FORTRAN IV (H) Programmer's Guide Re: C28-6602 with N28-2211	NEW
N28-2229	IBM S/360 Operating System: System Control Blocks Re: C28-6628 with N28-2193 & N28-2196	NEW
N28-2230	IBM S/360 Operating System: System Generation Re: C28-6554-2	NEW
N28-2231	IBM S/360 Operating System: Storage Estimates Re: C28-6551-3	NEW
N28-2235	IBM S/360 Operating System: Messages, Completion Codes & Storage Dumps Re: C28-6631-1	NEW
N28-2238	IBM S/360 Operating System: Sort/Merge Re: C28-6543-3 with N28-2216	NEW
N30-5006	TNL to OS QTAM Message Control Program Re: C30-2002-1	NEW
N30-5005	TNL Re: C30-2003-1 to OS QTAM Message Processing Program Services	NEW
N30-5007	TNL Re: C30-2004-0 to IBM S/360 Operating System: Basic Telecommunications Access Method	NEW
R20-4043-1	IBM S/360 O.S. Coding Education Guide	Scrap R20-4043-0
R20-8015-3	IBM S/360 Mod 20 EPIC - Problems and Case Study Part I	Use R20-8015-2 & N23-0007
R20-9203-0	IBM 2260 Display Station Course Description	NEW
V25-6267-0	IBM S/360 STR Line Control Magnetic Sets	NEW
Y20-0027-0	DP Application Program Standards Manual	Revised Abstract
Y26-8002-0	IBM 1800 Time-Sharing Executive System PLM Re: Y26-3702-0	NEW
Z20-1735-1	2400 Magnetic Tape Performance Considerations and Characteristics	Scrap Z20-1735-0
Z32-0400-1	IBM Computer Systems Training Student Handouts	Scrap Z32-0406-0, Z32-0400 & Z32-0409
Z77-7122	Customer Information System File Concepts for a Public Utility Company	NEW
Z77-7123	Conversion Aids for System/360	NEW
Z77-7124	Modified D.E.B.E. for System 360's Without a Console Typewriter	NEW

- PRL #21 May 26, 1967 -

N20-0030-11	**Catalog of Programs for IBM System/360 - May 1967 Supplement to C20-1619-2**	**Scrap N20-0030-10**
N20-0031-7	**Catalog of Program for IBM 1130 Computer System and IBM 1800 Data Acquistion and Control System - May 1967 Supplement to C20-1630-1**	**Scrap N20-0031-6**

MARKETING PUBLICATIONS

320-0935-1	**Application Guide Public Accounting**	**Use 320-0935-0**
320-1906-4	**In Brief, April 1967**	**Scrap 320-1906-3**
322-0033	**IBM Journal of Research and Development, Volume 11, Number 2**	**NEW**
560-0108-2	**IBM Reed Relays**	**Scrap 560-0108-1**
C20-1671-0	**Applications and Techniques of Operations Research ... A Selected Bibliography**	**NEW**
C24-3361-5	**IBM S/360 BOS Assembler with I/O Macros Specifications**	**Scrap C24-3361-4 ***
C24-3418-2	**IBM S/360 BPS/RPG (Tape) Specifications**	**Use C24-3418-1 ****
C24-5026-2	**IBM S/360 BPS Card & Tape Utility Programs Specifications SRL Bulletin**	**Use C24-5026-1 & N21-5030-0**
C28-6630-1	**IBM S/360 Operating System: Starter Operating System Guide**	**Scrap C28-6630-0, N28-2192, N28-2197, N28-2199 & N28-2215**
C30-2005-0	**IBM S/360 Operating System: Queued Telecommunications Access Method: Message Control Program Specifications for a Future Version**	**NEW**
C50-0001	**IBM 2701 Data Adapter Unit and ASCII Autodin Adapter RPQ F16124**	**NEW**
E20-0252-0	**Public Utilities - Construction Management System/360 DP Application Manual**	**NEW**
E20-0283-0	**Scheduling Empty Freight Cars on the Louisville and Nashville Railroad**	**NEW**
H20-0229-1	**1404 Printer Public Utilities DP Application Bulletin**	**Scrap H20-0229-0**
H20-0249-1	**1130 Work Measurement Aids (1130-MF-03X) Application Description**	**Use H20-0249-0 & N20-0076**
J20-0011-0	**Introduction to DATATEXT**	**NEW**

- PRL #21 (continued) -

K20-0212-0	Online Mortgage and Savings Accounting at Community Federal Savings and Loan Association	NEW
K20-0217-0	A Gauge Recall System at the Bendix Fluid Power Division with the IBM 357 Data Collection System	NEW
N22-0264	Changes to IBM 2400 Magnetic Tape Units Original Equipment Manufacturer's Information Re: A22-6862-3	NEW
N23-0604-0	ENL to IBM S/360 COBOL Prog. P.I. Text Re: R29-0215-0	NEW
N26-0183-0	IBM 1800 DP I/O TNL to A26-5969-3	NEW
N30-5004-0	IBM S/360 Operating System: Queued Telecommunications Access Method: Message Processing Program Services Re: C30-2003-1 with N30-5005	NEW
N30-5013-0	IBM S/360 Operating System: Basic Telecommunications Access Method Re: C30-2004-0	NEW
R20-4044-1	IBM S/360 O.S. Coding Student Materials	Scrap R20-4044-0
R20-4085-0	IBM 2260 Display Station Education Guide	NEW
R20-4086-0	IBM Data Communications Concepts Education Guide	NEW
R20-9204-0	IBM Data Communications Concepts Course Description	NEW
R29-0082-3	FORTRAN IV for IBM System/360 P.I. Chapter 2	Use R29-0082-2
R29-0090-1	IBM S/360 RPG Coding Advisor Guide Supplement	Use R29-0090-0
R29-0106-0	FORTRAN for the IBM 1130 P.I Problem Book	Use R29-0106-0
R29-0145-2	IBM PCDPP P.I. 85 Collator Text	Use R29-0145-1
*With C24-3361-4 scrap N24-5113, N24-5137, N24-5153, N24-5117, N24-5166, N24-5179 & N24-5270.		
** With C24-3418-1 use N24-5056, N24-5160, N24-5176 & N24-5200.		
R29-0199-2	IBM PCDPP Basic Machine Operation and Wiring Notebook Programmed Instruction Course	Use R29-0199-1
R29-1260-2	IBM Oper. the 1260 Elect. Inscriber Self-Study Course	Scrap R29-1260-1
X20-1701-1	S/360 GPSS Coding Form	Use X20-1701-0
X20-1748-1	DOS/TOS Job Control Language & Operator Communication Reference Card	Scrap X20-1748-0
X26-5508-0	IBM S/360 Process I/O Devices Template	NEW
Y20-0069-0	S/360 Attached Support Processor System (ASP) (360A-CX-15X) System Manual	NEW
Y24-5061	TNL to IBM S/360 Basic Operating System, System Control PLM Re: Y24-5002-0	NEW
Y27-7113-1	IBM S/360 Operating System Graphics Access Method PLM	Scrap Y27-7113-0 & Y27-7137-0
Y27-7141-0	IBM S/360 Operating System: Graphic Problem Oriented Routines Re: Y27-7110-0 PLM	NEW
Y28-2174-0	IBM S/360 Operating System: Fixed Task Supervisor PLM Re: Y28-6612-0, -1, -2 with Y28-2141 & Y28-2161	NEW
Y28-2208-0	IBM S/360 Operating System: Linkage Editor PLM Re: Y28-6610-1 with Y28-2175	NEW
Y28-2222-0	IBM S/360 Operating System: Introduction to Control Program Logic Manual Re: Y28-6605-2, with Y28-2180 & Y28-2203	NEW
Y28-2232-0	IBM S/360 Operating System: Input/Output Supervisor, PLM Re: Y28-6616-0 with Y28-2155, Y28-2169 & Y28-2181	NEW
Y28-2233-0	IBM S/360 Operating System: Utilities, Program Logic Manual Re: Y28-6614-0, -1 with Z28-2124, Y28-2154, Y28-2163 & Y28-2179	NEW
Y28-2234-0	IBM S/360 Operating System: FORTRAN IV (H) PLM Re: Y28-6642-0	NEW
Y28-6383	IBM S/360 Operating System FORTRAN IV (E) PLM Re: Y28-6601-2	NEW
Y28-6384	IBM S/360 Operating System: FORTRAN (G) PLM Re: Y28-6638-0	NEW
Y28-6613-1	IBM S/360 Operating System: Job Management, PLM	Scrap Y28-6613-0, Y28-2191 & Y28-2184
Y30-2001-1	IBM S/360 Operating System: Basic Telecommunications Access Method, PLM	Scrap Y30-2001-0
Y30-2003-0	IBM S/360 Operating System: Queued Telecommunications Access Method, PLM Re: Y30-2002-0	NEW
Y30-5002-0	IBM S/360 DOS QTAM PLM	NEW
Y30-5003-0	TNL to DOS BTAM PLM Re: Y30-5001-1	NEW
Z20-0835-0	Current Systems FORTRAN IV - System/360 FORTRAN Difference Manual Re: Z20-1715-0	NEW
Z77-7129-0	System/360 Direct Access Organization and Retrieval Techniques - Course Outline	NEW
Z77-7130	System Language 1 (SL/1) - A Procedural Programming Language in Operating System/360 (OS/360) Assembly Macros	NEW
Z77-7131	OS DD Reference Sheet	NEW
Z77-7132	COBOL F Timing and Coding Tips	NEW
Z77-7133	A Method of Interpolating an Empirical Graph on a System/360 Model 20 Using RPG as a Source Language	NEW
Z77-7134	Control of Pulp Bleach Plant	NEW

SECOM - NEW FEATURES

In the process of developing and improving SECOM to meet the suggestions and needs of Systems Engineers, new features are being added. Four new features are announced in this article.

1. Type II Programs

   All available information on Type II Programs has been added to the SECOM file. This includes APAR's and other temporary fixes as well as pointers, restrictions, documentation changes, etc. Each Industry Development group is responsible for maintaining the file of information in this system for their programs.

2. 1800 and 1130

   Arrangements have been made to have information needed by the SE's in using the 1800 and 1130 systems co-ordinated in San Jose. The same type of information will exist as does for S/360 Type I Programs. Input is coordinated by the SECOM Coordinator, Department 954, San Jose Plant.

3. Improved ITPS Retrieval

   Improved turn-around time for ITPS retrieval is now possible with the implementation of direct retrieval by the ITPS facility in Armonk, New York. ITPS operators have the required instructions. Turn-around of 10 to 45 minutes has been experienced in most cases.

4. Current Level Reporting

   Several types of programs, documentation, and Engineering Change level functions are in this system; not new but improved. One function that may be used now covers the Engineering Change levels required for various functions of each system. It is under the program information (PIN) section of RETAIN and usually cataloged with the program number of the control program and the suffix "ec." For example, the request pin-360n-cl-453-ec would provide the significant EC levels for DOS. Note that lower case is used for requests.

   Another function that may be used now covers the level of programs, SRL's and PLM's for DOS, TOS and BOS. The request options are:

   bos pgm
   pin 360 tos srl
   dos plm

   For example: the request **pin-360-dos-srl would give you the current level of** DOS-SRL's and any TNL's needed to update them.

Any suggestion or comments on SECOM will be appreciated. They should be sent to the DP SECOM/RETAIN Coordinator, Systems Engineering Technical Programs, DPD HQ, 112 East Post Road, White Plains, New York 10601.

PID ORDER STATUS INQUIRY SERVICE

In certain situations, it is imperative to know the status of an order for a program submitted to PID (DP Program Information Department). PID can provide this information rapidly in response to a phone call (914-592-5790) or an ITPS wire.

To save time, be sure you have all the necessary information before you call or wire. Certain details are vital for tracing an order. Some of these are:

1. Customer number, name and shipping address.
2. Branch Office number.
3. Program order number.
4. Material ordered (basic or optional).
5. Was maintenance requested?
6. Date order was submitted.
7. How order was submitted; for example

   - Was a magnetic volume (tape/disk) involved?
   - Mode of Shipment
     United Parcel Service, Air Freight, etc.
     U.S. Mail - First Class, Parcel Post, Insured, etc.
   - Number of tapes or disk packs -- how many physical packages?
   - Was the program order form enclosed with the volumes?
   - Was a magnetic tape ordered? If so, was form #170-1138 used? Was it sent with the program order form? Were they sent to PID or to Minneapolis?
   - Were other program orders involved? What were the programs?
   - Was any special handling requested?

Inquiries will reach the appropriate PID coordinator if you specify system type and trans-mittal media (example: S/360 card program

coordinator). In case any follow-up is necessary, make a note of his name.

Use the Inquiry Service with discretion since excessive use will adversely impact the PID processing turnaround time for normal orders.

INFORMATION ABOUT THE NEWSLETTER

EVERY SALESMAN AND SYSTEMS ENGINEER SHOULD RECEIVE ONE COPY OF THE NEWSLETTER. Each FE Branch Manager should receive five copies for distribution to Customer Engineers.

The IBM Installation Newsletter is distributed only to IBM locations and is not available to customers. The Newsletter is separated into two parts. The first part contains information which is intended only for IBM personnel and the pages are labeled For IBM Internal Use Only. THESE PAGES, OR REPRODUCTIONS OF THEM, ARE NOT TO BE GIVEN TO CUSTOMERS. However, using your own good judgement, you may discuss pertinent information from these pages with a customer. The second part contains information which may be REPRODUCED by the Branch Office at their discretion and given to customers. These pages are not labeled with any restrictive classification. Note that when any abstracts for Type III programs are given to the customer, he must be informed that the disclaimer for Type III programs, as contained in the Newsletter, applies. YOUR COOPERATION IN PROPERLY HANDLING THE CONTENT OF THE IBM INSTALLATION NEWSLETTER IS NECESSARY FOR ITS CONTINUED PUBLICATION.

Changes to the quantity of the Newsletter shipped are to be made by the Administration Manager. (NOTE THAT EXTRA COPIES OF THE NEWSLETTER ARE NOT TO BE ORDERED FOR DISTRIBUTION OF INFORMATION TO CUSTOMERS.) He should update and send the DP Administration Managers' Communication Control card (form number N44-780) to DP Communication Control, DPD HQ. (The card N44-780 replaces N37251.)

Direct inquiries concerning the contents of this Newsletter to:

Editor
IBM Installation Newsletter
Systems Engineering Technical Programs
DPD HQ
112 East Post Road
White Plains, New York 10601
Telephone 914 WH9-1900 Extension 4781

June 16, 1967 Issue No. 67-12

IBM INSTALLATION NEWSLETTER

TABLE OF CONTENTS

OS/360 RELEASE 11 DLIB CHANGE

Effective with Release 11, the normal distribution of full OS/360 from PID will consist of DLIB01, DLIB02 and DLIB03. MODLIB has been removed from DLIB01 and is now the sole DATASET on DLIB03. (Note that this change does not apply to Customized Distributions.) The corresponding volume serial changes are required in both the JCL used for SYSGEN, and any PTF's carried over from Release 10 if they are to be applied to MODLIB.

OS/360 JCL CHANGE FOR 1600 BPI TAPE UNITS

DD cards with UNIT=2400 reference only the 2400 models 1, 2 or 3. Users of 2400 models 4, 5 and 6 must specify UNIT=2400-3 for 800 BPI or 2400-4 for 1600 BPI. For a device and model independence SYSGEN, a user can specify an assigned name (example, UNIT=TAPE).

An installation with 2400 Model 4, 5 or 6 only, may SYSGEN a user assigned name of USERNAME=2400.

OS/360 - JCL ACCOUNTING DATASET ABEND FIX

The Operating System (Release 11) will ABEND (F23) under the following circumstances.

1. The accounting dataset (SYS1.ACCT) is not correctly specified at IPL time.

2. The accounting dataset fills up during execution.

This can be remedied by using SUPERZAP (Type III described in Newsletter issue 67-11) with the following control cards.

1. The SYSLIB card. DSNAME=SYS1.SVCLIB

2. SUPERZAP Control Cards

```
DUMPT   IGC0003E  ALL
NAME    IGC0003E
VERIFY  16   4740
REP     16   4700
DUMPT   IGC0003E  ALL
```

OS/360 - RELEASE 11 ACCOUNTING ROUTINE

This accounting routine provides the user with the ability to write accounting data on a resident direct access device. The writer assembly module (IEFWAD), is more or less an access method which writes and controls the data set SYS1.ACCT. It is a device dependent module which resides in SYS1.LINKLIB and uses a ten byte resident nucleus work area. A sample accounting routine IEFACTRT has been stored in SYS1.SAMPLIB where it is named SAMACTRT.

1. Initialization

   The facilities to use this accounting routine are included in any system for which ACCTRTN SUPPLIED is specified at SYSGEN. However, the sample accounting routine, IEFACTRT, which contains the linkage to IEFWAD and thereby triggers this accounting facility, exists only on SYS1.SAMPLIB where it is named SAMACTRT and is not included in the SYSGENed system. Therefore, as is the case now, an accounting routine which satisfies user requirements may be built into the system. Whether or not this routine uses the facility provided by IEFWAD is an option of the user.

   The data set to be written must:

   a. Be pre-allocated on a permanently resident DA device.
   b. Have the name SYS1.ACCT.
   c. Have no secondary extent.
   d. Be allocated contiguous space.

2. Sample Account Routine (IEFACTRT)

   Input

   a. Register 0 contains a code, indicating at what time the routine was called;

      code: 8 Step Initiation
            12 Step Termination
            16 Job Termination

   b. Register 1 contains the address of a list of 4-byte pointers to:

      job name
      *step name
      programmers name
      job running time

job accounting data
*step running time
*step accounting data
flags (byte 1), step number (byte 2)

*The asterisked pointers will be 0 at Job Termination. If the job has been cancelled, bit 7 of flags is set to 1.

c. Registers 13, 14 and 15 are used as specified by OS/360 linkage conventions.

3. Entrances to Accounting Routine

a. Job Initiation = Step initiation for first step
b. Step Initiation
c. Step Termination
d. Job Termination

4. Output

a. A line on SYSOUT containing date and time of day in hours with four decimals. The description by column reference, Job Initiation, Step Termination, and Job Termination, respectively is shown in Exhibit 1.

b. A record to be written in the Account Data Set as shown in Exhibit 2.

At step initiation there will be output only if the step number is 1, i.e. at Job Initiation.

If at Step Initiation, the first byte of job account data is X'FF', bit 7 of flags is set to 1 and the job will be cancelled. Any user who desires to use his accounting routine to determine if a job is to be run may do so and a cancel will be effected if the accounting routine sets bit 7 of flags to 1.

If the account data cannot be written in the account data set, the SYSOUT message giving date and time is written to the console. To the right of this message, a one digit code indicates the reason for the failure as follows:

Code Interpretation

1) Record is too long to fit on a single track.
2) No space left in SYS1.ACCT.
3) SYS1.ACCT not found.
4) Permanent I/O error.
5) End of file not found.
6) Unit not found.

5. Account Data Set Writer (IEFWAD)

a. Input

1) Register 1 contains the address of the record to be written.

Format:			
	DS	3H	space to be used by this module
	DC	H'__'	contains data length in bytes. Cannot exceed 1 track.
	DC	___	user data. Data which is written to SYS1.ACCT.

2) Registers 13, 14 and 15 are used as specified by OS/360 linkage conventions.

b. Function

If it is the first entry after IPL, or the first entry after a SET ACCT command, or the reply to an IEF506D message, the data set SYS1.ACCT is OPENED before writing into it. Thereafter, track balance, TTR for the next record write and TTR for the last track of SYS1.ACCT extent are maintained in core, eliminating any need for subsequent OPENS. The last record written to SYS1.ACCT is always an end of file with the key equal to the track balance. In the circumstance where a warm start is required, this end of file may be used to locate the last record written to SYS1.ACCT. Provided only that there is space remaining in the extent of SYS1.ACCT, each record of account data is appended to the data set. To direct account data to be written from the start of SYS1.ACCT, N (new) may be specified in the SET ACCT command or in the reply to an IEF506D message. In effect, specifying N deletes all prior data of SYS1.ACCT and renders the allocated space reusable.

c. Console Messages are generated as follows:

IEF506D SYS1.ACCT, XXX, N	SPACE NOT AVAILABLE NOT PREALLOCATED PERMANENT I/O ERROR

ERROR IN UNIT NAME
SYNTAX ERROR
NO END OF FILE

The operator reply then is entered:

id IEF506I REPLY xxx or xxx, N or SKIP, xxx=Device name

where;

xxx is a device name and N designates use of a new SYS1.ACCT, i.e. the account data should be written from the beginning of the SYS1.ACCT extent. When N is not specified a warm start will be attempted.

1) Space Not Available

Explanation: the extent of SYS1.ACCT has been filled.

Action

a) If a data set named SYS1.ACCT has been allocated to two resident devices, the operator may reply with the unit name of the second device. N is used to designate that the writer may start at the low extent. A job should be submitted to retrieve the data from the SYS1.ACCT data set which is full, so that it in turn may be used when the second data set is full.

b) If multiple SYS1.ACCT data sets are not available for use, the operator may reply SKIP in which case Date and Time of Day with a reason code is dumped to the console. A job to retrieve data from the SYS1.ACCT data set should be submitted. When the job is complete the operator may reply to the next occurrence of this message with XXX, N which tells the system that the space allocated to SYS1.ACCT is again available for use from its start.

2) Not Preallocated

Explanation: A data set named SYS1.ACCT is not found on the device specified, or, if no device has been specified, on the system residence device.

Action

a) If SYS1.ACCT resides on a device other than that specified, reply with device on which SYS1.ACCT resides. N specifies that the account data is to be written from the beginning of SYS1.ACCT.

b) If SYS1.ACCT resides on no device, reply SKIP and run a job allocating SYS1.ACCT.

3) Permanent I/O Error

Explanation: A permanent I/O error occurred while processing SYS1.ACCT; consequently the program is unable to further process this data set.

Action

a) If a data set named SYS1.ACCT has allocated to two resident devices, the operator may reply with the unit name of the second device. N specifies that the account data is to be written from the beginning of SYS1.ACCT.

b) If there is no such device reply, SKIP and run a job allocating SYS1.ACCT to another device. When this is not possible a job may be submitted to retrieve the SYS1.ACCT data. When this job is complete, the operator may reply to the next occurrence of this message with XXX, N which tells the system that the space allocated to SYS1.ACCT is again available for use from its start.

4) Error in Unit Name

Explanation: An invalid unit name has been specified.

Action: Specify the correct unit name.

5) Syntax Error

Explanation: Syntax error in IEF506I reply message.

Action: Re-submit reply.

6) No End of File

Explanation: The end of file terminating the account data set has been lost; consequently, the program is unable to position to the end of this data set to write the next record.

Action

a) If a data set named SYS1.ACCT has been allocated to two resident devices, the operator may reply with the unit name of the second device. N specifies that the account data is to be written from the beginning of SYS1.ACCT.

b) If there is no such device reply, SKIP and run a job allocating SYS1.ACCT to another device. When this is not possible a job may be submitted to retrieve the SYS1.ACCT data. When this job is complete, the operator may reply to the next occurrence of this message with XXX, N which tells the system that the space allocated to SYS1.ACCT is again available for use from its start.

In every circumstance where the reply SKIP is used, the console message will be repeated at the next entrance to the accounting routine whenever the factors which caused the original message still prevail.

d. Output

Register 0 The number of empty tracks in SYS1.ACCT

Register 15 Condition codes

code	interpretation
0	Normal exit
4	Record to be written is longer than a track
8	No space left in SYS1.ACCT
12	SYS1.ACCT not found
16	Permanent I/O error
20	SYS1.ACCT end of file not found
24	Unit name not found

e. Use of ENQ/DEQ

IEFWAD enques on the major Q name SYSIEFAR and the minor Q name WD.

f. Specifying the Device on which SYS1.ACCT Resides

The parameter ACCT = (unitname , N) has been added as an option to the SET command where: unit name specifies the device on which SYS1.ACCT resides. If this parameter is omitted, the system residence volume is assumed, and N specifies that the lowest extent of SYS1.ACCT may be used. If this parameter is omitted, a warm start will be attempted from the last record written.

OS/360 - RELEASE 11 ACCOUNTING ROUTINE

7-10	13-20	22-26	35-38	39-44	51-61	63-69
JOB	jobname	START	DATE	YY.ddd	Time of Day	hh.ffff
	stepname	END	DATE	YY.ddd	Time of Day	hh.ffff
JOB	jobname	END	DATE	YY.ddd	Time of Day	hh.ffff

where
YY.ddd is the year and Julian day

and
hh.ffff is the time in hours with four decimals.

Exhibit 1

OS/360 - RELEASE 11 ACCOUNTING ROUTINE

Description

BYTE	Code =8	Code =12	Code=16
1-3	ACT	ACT	ACT
4	X'8'	X'C'	X'10'
5-12	Job name	Job name	Job name
13-16	Date (packed)	Date	Date
17-20	Time (packed)	Time	Time
21-28	Pro-	Step name	Number of steps (1 byte)
29-36	grammer	Program name	
37-40	name	Completion code	
41		step number	
42	Account data from JOB card	Account data from EXEC card	

Exhibit 2

OS/360 TECHNIQUE TO TYPE COMPLETION CODES ON CONSOLE

The following technique has been developed for use in an OS/360 installation and contributed to the Newsletter. Users should be aware that it has not been submitted to any formal IBM test and that no maintenance support will be available. The technique has been used successfully to type System Completion Codes on the Console Typewriter. The purpose is to inform the operator, on the console, of the cause of termination. It provides the Return Code on Normal Step Termination and the ABEND Code on Abnormal Termination. The listing and comments (Exhibit 1) should be self-explanatory to the user. The contributor has used the technique successfully with OS/360 Releases 9 through 11. It is especially useful for checking out new programs and Job Control Language.

OS/360 TECHNIQUE TO TYPE COMPLETION CODES ON CONSOLE

COMPLETION CODE TYPE OUT APPROACH FOR OS/360

THE FOLLOWING APPROACH HAS BEEN SUCCESSFULLY USED TO TYPE SYSTEM COMPLETION CODES TO THE CONSOLE.

THIS IS ACCOMPLISHED BY:

1. COMPILING THE SYSTEM CODE ANALYSIS ROUTINE (CODE PROVIDED). THIS ROUTINE BECOMES PART OF THE SCHEDULER VIA STEP 2.

2. LINKEDIT THE MODULE INTO THE SCHEDULER IN MODLIB PRIOR TO STAGE TWO OF SYSTEM GENERATION (ILLUSTRATION PROVIDED).

3. MODIFYING THE MACRO SGIEF442 IN GENLIB PRIOR TO STAGE ONE OF SYSTEM GENERATION (ILLUSTRATION PROVIDED).

Exhibit 1

OS/360 TECHNIQUE TO TYPE COMPLETION CODES ON CONSOLE

```
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
+++++
+ 1.  SOURCE CODE OF SYSTEM COMPLETION CODE ANALYSIS ROUTINE
+++++


         TITLE 'DPHQ MODIFIED IEFW42SD FOR ERROR CODE ANALYSIS'          00000000
IEFW42SD CSECT                                                           00000100
*************************************************************************00000200
*************************************************************************00000300
*                                                                       *00000400
*                                                                       *00000500
*                                                                       *00000600
*                                                                       *00000700
*  MODULE NAME IEFW42SD                                                 *00000800
*                                                                       *00000900
*                                                                       *00001000
*        ENTRY: STEP TERMINATION  FROM MODULE IEFSD011                  *00001100
*                                                                       *00001200
*                                                                       *00001300
*        EXIT: TO DPQW42SD (ORIGINALLY IEFW42SD RENAMED FOR DPHQ)       *00001400
*                                                                       *00001500
*                                                                       *00001600
*        THIS MODULE ANALYZES THE CAUSE OF TERMINATION AND INFORMS      *00001700
*        THE OPERATOR OF THE RETURN CODE ON NORMAL STEP TERMINATION     *00001800
*        OR THE ABEND CODE ON ABNORMAL TERMINATION                      *00001900
*                                                                       *00002000
*                                                                       *00002100
*        INPUT: PARAMETER LIST IN REGISTER #1 CONTAINING THE            *00002200
*        ADDRESSES OF THE LCT,THE JCT, AND THE SCT. ADDITIONAL          *00002300
*        INPUT INFORMATION IS SUPPLIED THROUGH THECVT WHICH SUPPLIES     *00002400
*        TCB ADDRESS FOR THE TERMINATED TASK. ALTHOUGH ALL OF THESE     *00002500
*        CONTROL BLOCKS ARE NOT PRESENTLY USED BY THE MODULE,           *00002600
*        DSECT ADDRESSING HAS BEEN SUPPLIED TO MANY TO ALLOW FOR        *00002700
*        FUTURE EXPANSION                                               *00002800
*                                                                       *00002900
*                                                                       *00003000
*        OUTPUT: MESSAGES TO OPERATOR. THE ORIGINAL CONTENTS OF         *00003100
*        ALL REGISTERS IS RESTORED BEFORE BRANCHING TO DPQW42SD         *00003200
*                                                                       *00003300
*                                                                       *00003400
*                                                                       *00003500
*                                                                       *00003600
*                                                                       *00003700
*                                                                       *00003800
*                                                                       *00003900
*                                                                       *00004000
*************************************************************************00004100
*************************************************************************00004200
         EJECT                                                           00004300
R0       EQU   0                                                         00004400
R1       EQU   1                                                         00004500
R2       EQU   2                                                         00004600
R3       EQU   3                                                         00004700
R4       EQU   4                                                         00004800
R5       EQU   5                                                         00004900
R6       EQU   6                                                         00005000
R7       EQU   7                                                         00005100
R8       EQU   8                                                         00005200
R9       EQU   9                                                         00005300
R10      EQU   10                                                        00005400
R11      EQU   11                                                        00005500
```

Exhibit 1 (continued)

OS/360 TECHNIQUE TO TYPE COMPLETION CODES ON CONSOLE

```
R12      EQU   12                                                    00005600
R13      EQU   13                                                    00005700
R14      EQU   14                                                    00005800
R15      EQU   15                                                    00005900
         BALR  R9,R0              ESTABLISH                          00006000
         USING *,R9              ADDRESSABILITY                      00006100
         ST    R1,IEFONES            SAVE PARAMETER LIST ADDRESS     00006200
         STM   R2,R8,IEFTWSVN        SAVE WORKING REGISTERS          00006300
         L     R5,16(0,0)          PLACE CVT ADDRESS IN REGISTER 5   00006400
         L     R3,0(0,R5)            AND PICK UP                     00006500
         L     R3,4(0,R3)          TCB ADDRESS                       00006600
         USING JCT,R6                                                00006700
         USING SCT,R7                                                00006800
         USING LCT,R8                                                00006900
         USING TCBDSECT,R3                                           00007000
         TM    TCBFLAGS,X'80'          TEST FOR ABEND                00007100
         BC    1,CONCDANL            YES, GO TO CONDITION CODE ROUTINE 00007200
         L     R8,0(0,1)           LOAD LCT ADDR INTO REG 8          00007300
         MVC   LCTJCTAD(4),4(1)                                      00007400
         MVC   LCTSCTAD(4),8(1)                                      00007500
         LM    R6,R7,LCTJCTAD  GET ADDRESS OF JCT AND SCT IN CORE    00007600
         LH    R4,TCBCOMP+2          GET RETURN CODE INTO REG 4      00007700
         N     R4,CONFMASK                                           00007800
         CVD   R4,DOUBLWRD           CONVERT RETURN CODE TO DEC      00007900
         OI    DOUBLWRD+7,X'0F'            SET PROPER ZONE           00008000
         UNPK  DOUBLWRD(4),DOUBLWRD+5(3)     UNPACK USER CODE        00008100
         MVC   PRINTOUT+11(4),DOUBLWRD   MOVE RC INTO OUTPUT AREA    00008200
PRINTOUT WTO   'RC=     '                                            00008300
         BC    15,EXIT                                               00008400
CONCDANL L     R4,TCBCOMP                                            00008500
         N     R4,SYSTMASK                                           00008600
         C     R4,SYSTCHEK           CHECK FOR USER ABEND            00008700
         BH    SYSTCODE      NO? GO TO SYSTEM ABEND ERROR ANALYSIS   00008800
         LH    R4,TCBCOMP+2          GET USER COMPCODE               00008900
         N     R4,CONFMASK            REMOVE EXTRANEOUS INFOR        00009000
         CVD   R4,DOUBLWRD            CONVERT USER CODE TO DECIMAL   00009100
         OI    DOUBLWRD+7,X'0F'                                      00009200
         UNPK  DOUBLWRD(4),DOUBLWRD+5(3)     UNPACK USER CODE        00009300
         MVC   PRINTCOD(4),DOUBLWRD       MOVE INTO PRINT AREA       00009400
         MVI   PRINTCOD-1,C'U'          INDICATE TYPE                00009500
         BC    15,WTOROUTN                                           00009600
SYSTCODE UNPK  DOUBLWRD(5),TCBCOMP+1(3)     UNPACK HEX SYSTEM CODE   00009700
         TR    DOUBLWRD(3),TCBHEXL-240   TRANSLATE HEX FOR PRINTING  00009800
         MVC   PRINTCOD+1(3),DOUBLWRD                                00009900
         MVI   PRINTCOD-1,C'S'          INDICATE TYPE                00010000
WTOROUTN MVC   ERRTYPES+13(5),PRINTOT   MOVE IN FINAL MESSAGE        00010100
ERRTYPES WTO   '**CC=     **'                                         00010200
EXIT     L     R1,IEFONES          PICK UP PARM LIST ADDR FOR DPQ42SD 00010300
         LM    R2,R8,IEFTWSVN            RESTORE WORKING REGISTERS   00010400
         L     R15,ACONSTNT                                          00010500
         BCR   15,R15                                                00010600
         DS    0D                                                    00010700
DOUBLWRD DS    D                                                     00010800
IEFONES  DS    CL4                                                   00010900
IEFTWSVN DS    CL32                                                  00011000
CONFMASK DC    X'00000FFF'                                           00011100
ACONSTNT DC    V(DPQW42SD)                                           00011200
PRINTOT  DS    0CL5                                                  00011300
         DC    CL1' '                                                00011400
PRINTCOD DC    CL4'  '                                               00011500
CDOCONCD DC    2F'0'                                                 00011600
SYSTMASK DC    X'00FFFFFFFF'                                         00011700
SYSTCHEK DC    F'4095'                                               00011800
         DS    CL8                                                   00011900
```

Exhibit 1 (continued)

OS/360 TECHNIQUE TO TYPE COMPLETION CODES ON CONSOLE

```
TCBHEXL  DC     C'0123456789ABCDEF'                                    00012000
SCT      DSECT                                                         00012100
         DS     0D                                                     00012200
         DS     CL3                                                    00012300
         DS     CL1                                                    00012400
SCTSSTAT DS     CL1             INTERNAL SYSTEM STEP STATUS            00012500
         DS     CL3                                                    00012600
SCTSEXEC DS     CL2                                                    00012700
         DS     CL108                                                  00012800
SCTSDPCD DS     CL2                                                    00012900
         DS     CL4                                                    00013000
LCT      DSECT                                                         00013100
         DS     0D                                                     00013200
         DS     CL4                                                    00013300
         DS     CL4                                                    00013400
         DS     CL4                                                    00013500
         DS     CL4                                                    00013600
LCTJCTAD DS     CL4                                                    00013700
LCTSCTAD DS     CL4                                                    00013800
         DS     CL4                                                    00013900
         DS     CL4                                                    00014000
         DS     CL4                                                    00014100
         DS     CL4                                                    00014200
         DS     CL4                                                    00014300
         DS     CL4                                                    00014400
         DS     CL4                                                    00014500
LCTCMCBA DS     CL4                                                    00014600
TCBDSECT DSECT                                                         00014700
         DS     0D                                                     00014800
         DS     3F                                                     00014900
TCBTID   DS     F                                                      00015000
TCBCOMP  DS     F                                                      00015100
         DS     CL9                                                    00015200
TCBFLAGS DS     CL5                                                    00015300
         DS     CL6                                                    00015400
TCBJLB   DS     CL4                                                    00015500
JCT      DSECT                                                         00015600
         DS     0D                                                     00015700
         LTORG                                                         00015800
         END                                                           00015900
```

```
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
+++++
+ 2.  ALTER SCHEDULER TO INCLUDE ANALYSIS ROUTINE.
+       NOTE: MUST PERFORM PRIOR TO STAGE TWO OF SYSTEM GENERATION.
+++++


//MODSCHED JOB 2,'MOD SCHEDULER',MSGLEVEL=1
//STEPLKED EXEC PGM=IEWL,PARM='LET,XREF,LIST'
//SYSLIB DD DSNAME=SYS1.MODLIB,DISP=OLD
//SYSLMOD DD DSNAME=SYS1.MODLIB,DISP=OLD
//SYSUT1  DD DSNAME=SYS1.UT3,DISP=OLD
//SYSPRINT  DD SYSOUT=A
//SYSLIN DD *
 CHANGE IEFW42SD(DPQW42SD)
  INCLUDE SYSLIB(IEFW42SD)
 NAME DPQW42SD(R)
***** INCLUDE OBJECT CODE OF MODIFIED IEFW42SD HERE *****
 NAME IEFW42SD(R)
+++++ INSERT '/*' CARD HERE'
```

Exhibit 1 (continued)

OS/360 TECHNIQUE TO TYPE COMPLETION CODES ON CONSOLE

```
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
+++++
+ 3.  ALTER SYSGEN MACRO PRIOR TO STAGE ONE.
+        NOTE: CHECK STATEMENT NUMBERS BEFORE USING.
+++++


//CHGSGMAC JOB 3,'CHG SG MACRO',MSGLEVEL=1
//MOD EXEC PGM=IEBUPDAT,PARM=MOD
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSNAME=SYS1.GENLIB,DISP=OLD
//SYSUT2 DD DSNAME=SYS1.GENLIB,DISP=OLD
//SYSIN DD *
./        CHNGE SGIEF442,01,0,1
./        DELET 00260000,00260000
 PUNCH ' INCLUDE MODLIB(IEFSD011,IEFW42SD,DPQW42SD,IEFYNIMP)'          00260000
 PUNCH ' INCLUDE MODLIB(IEFYPJB3,IEFVJIMP)'                            00260100
./        DELET 05220000,05220000
 PUNCH ' INCLUDE MODLIB(IEFSD011,IEFW42SD,DPQW42SD,IEFYNIMP)'          05220000
 PUNCH ' INCLUDE MODLIB(IEFYPJB3,IEFVJIMP)'                            05220100
./        DELET 08980000,08980000
 PUNCH ' INCLUDE MODLIB(IEFSD011,IEFW42SD,DPQW42SD,IEFIDUMP)'          08980000
 PUNCH ' INCLUDE MODLIB(IEFYNIMP,IEFYPJB3)'                            08980100
./        ENDUP
+++++ INSERT '/*' CARD HERE


    PROCESSING ENDED AT EOD


IEF285I   SYSOUT                                        SYSOUT
IEF285I   VOL SER NOS=        .
IEF285I   SYSOUT                                        SYSOUT
IEF285I   VOL SER NOS=        .
SYS=65, JOB=LIST    , STP=EX       , PROJ=40001, PGMR=FVOSS, DATE=67/149,
SYS=65, JOB=LIST    , STP=EOJRLS11, PROJ=40001, PGMR=FVOSS, DATE=67/149,

TIME=16-43-32.9, RC=0000
TIME=16-43-34.6



//LIST JOB 40001,FVOSS,MSGLEVEL=1
//EX EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT2 DD SYSOUT=A,DCB=BLKSIZE=80
//SYSIN DD DUMMY,DCB=BLKSIZE=80
//SYSUT1 DD *
SYS=65, JOB=LIST    , STP=EX       , PROJ=40001, PGMR=FVOSS, DATE=67/149,
TIME=16-43-48.1, RC=0000
IEF236I ALLOC. FOR LIST     EX
IEF237I SYSUT1   ON 20C
```

Exhibit 1 (continued)

OS/360 BTAM I TO BTAM II CONVERSION

A Technical Newsletter (N30-5013) to the SRL "OS/360 BTAM" (C30-2004) is now available for ordering by the IBM Branch Office from the IBM Distribution Center, Mechanicsburg. This TNL contains an excellent write-up entitled "Requirements for Converting from OS/BTAM I to OS/BTAM II." Users converting from BTAM I in OS/360 Release 8 to BTAM II in Release 11 should carefully review this TNL and take appropriate action.

OS/360 QTAM RESTRICTIONS

1. OS/QTAM specifications limit the user to accessing a process queue through one DCB (job) only. However, no prevention is built into the current system to insure against this. If done, the last job initiated (OPENed) will function properly with any previous jobs left waiting once their Main-Storage queue is empty. The user would be unaware of this situation until his attempt to close fails to execute completely (message control will not terminate).

2. The following are two restrictions the user should insure against in using 'CONVERSE MODE' under current OS/QTAM support.

   a. When using conversation mode and the process queue option of 'EODAD' (empty queue exit), it is possible to freeze the source line indefinitely if the exit is taken. Until the exit is bypassed (by nature of finding an enqueued message), thereby initiating the 'GET' execution, the converse logic will not be entered to satisfy the input request.

   b. When converse mode is initiated in the user's Receive LPS, a check is made to insure that the destination of the message is a 'PROCESS' queue. No check is made, however, to determine if that queue has been opened by a process program. If it has not, the source line will be held in converse mode until that input is subsequently accessed and responded to by a message process program.

3. The current process program CLOSE will work properly providing it is done in conjunction with message control termination. The restriction limits the capability of closing a message process DCB with the intent of subsequently re-opening and continuing access of that process queue (example: rolling in and out of a low priority message job). If done, the next outstanding message in the queue will be lost and closely following entries to this queue could have unpredictable results.

   It is presently planned to remove all of the above restrictions during 4th Q 1967. Actual availability will be announced with a future Release or OS/360.

OS/360 QTAM CONVERSATION MODE DISCUSSION

The user has the capability of initiating 'MODE CONVERSE' on a communication line. By definition, QTAM will receive the entire message (up to EOT) and then freeze activity on that line until a response to that input has been made by a message process program. The only LPS initiated function which would reset a converse condition would be to CANCELM on the input message.

Access of this message is made through QTAM's 'GET' capability to enable the user's process function and response generation to take place. The response message 'PUT' is performed no differently than its normal enqueuing functions. Execution of the subsequent GET logic, which returns to message control the buffer used in satisfying the previous GET request, is the point in the cycle where a conversation condition is looked for. When converse is detected (the 2-bit on in LCBSTATE), the net result is a priority search of the source terminal's queue looking for a message from a process program. The first one found will be accessed and transmitted as the response. The user is vulnerable to the extent that any process program message found will satisfy the search with no guarantee that the message found is actually the desired response message (it is possible for other process programs to be PUTing messages to this same destination). The user could insure against this by placing a high priority on his converse response messages. If no process message is found in the queue, the conversation condition is reset and normal line activity resumes.

The conversation setting of the line will remain during transmission of the response and also at initiation of the subsequent receive operation. The theory is that this converse operation may extend beyond one input and response. Only a negative response to a poll will actually return the line to normal. The user could modify this logic by simply resetting the converse bit in the LCBSTATE field. If it were reset following the RCVHDR delimiter in the LPS (register 4 points to the LCB at this time), it would prevent subsequent transmissions from unconditionally being processed in converse mode. If it were reset following the SENDHDR delimiter in the LPS, it would prevent the unconditional return to polling at completion of sending the response and thereby allow normal line activity to be initiated (example - transmit other enqueued messages).

Dial Conversation

The conversation logic is no different for a dial environment than it is for leased lines. The dial connection is maintained until a response message is sent back to the source terminal. However, a unique problem does exist in the dial environment. When a message becomes enqueued for a dial terminal, message control will immediately start scanning those lines associated with its line group, searching for a free line which can be utilized. The enqueuing of a message as a result of a PUT will also trigger the search for a line even though a converse condition may exist. The net result is that if a line is available, the response message will be accessed and transmission attempted on other than the original line. When the converse condition is subsequently detected as a result of the next GET execution, no response message will be found and the connection broken. A further complication is that the alternate path will more than likely also fail by nature of the fact that the attempted dial will result in a "busy" condition. Only if no lines were found available on the search (all busy) will the response message still remain in the queue to enable normal execution of the desired conversation mode.

A possible solution would be to define Terminal Table entries representing line identities (as opposed to terminals) with a unique line group definition for each. The source and response destination of the converse message would then be the line with the input terminal more or less transparent to message control. It would require added user code in the LPS to resolve the line identifier (for converse conditions) to pass to the process program but the terminals could call in on any of the available lines. The normal terminal entries in the table would have to be associated with only one line group (and thus only one line) thereby limiting its other output to a single path. However, if converse is the primary environment, this limitation may not be significant.

OS/360 GRAPHICS

1. When the user receives control in his attention handling routine, the first byte of the COMAREA contains a unit number which indicates the 2260 which caused entry. This unit value, if to be used as the unit field (in a register) for an I/O macro, must be reduced by one. However, this applies only when there is more than one 2260 using the same DCB. If one DCB was set up for each 2260, this byte will always be zero.

2. The graphics SYSGEN macro statement will execute correctly only if a single residence system is created in OS/360 Release 11. If a split residence system with graphics is required, the graphics macro will attempt to create a SYS1.SVCLIB on the split volume during Stage 2. This can be corrected in three ways.

   a. Change the incorrect record in member SGIFF5LS of SYS1.GENLIB. On the fourth and fifth records, change 13 to 11 and 14 to 12 before SYSGEN.

   b. Change the COPY card in the IEHMOVE phase of Stage 2 in SYSGEN to reference SYS1.SVCLIB on the system residence volume instead of the split volume.

   c. IEHMOVE SYS1.SVCLIB from the split volume to the residence volume after SYSGEN.

OS/360 COBOL - LOOKING AHEAD TO MVT

If the user plans to go MVT (Multiprogramming with a Variable number of Tasks), he should be aware of the following considerations.

1. In MVT, all SYSIN data sets go first to direct access. If the COBOL problem program omits data set ORGANIZATION and ACCESS IS..., then QSAM is the access method used. DEVD, OPTCD, BUFNO, EROPT are data set attributes which may be left out of the problem program and merged from the DD card or DSCB.

   RECFM, LRECL, BLKSIZE, BFTEK, BFALN, BUFCB, EODAD, EXLST, SYNAD must be specified in the problem program. Once this is done, no override is possible from the DD card or the DSCB.

   Therefore, for SYSIN data sets, the problem program must be written to the same RECFM, LRECL, and BLKSIZE and the reader procedure IEFDATA card used. Since QSAM is used, two buffers are standard.

2. If the SORT verb is used, the region value must be sufficient to accommodate the load module plus a minimum of 17,048 bytes for sort program execution. If linkedit of modifications is to be done, then the region must be big enough to allow LINKEDIT.

3. Release 9.5 prose states ACCEPT...SYSIN, DISPLAY...SYSOUT, DISPLAY...SYSPUNCH and EXHIBIT may not refer to data sets specified as being blocked. Therefore, if these options are used in COBOL, IEFDATA must not be blocked in the reader procedure.

4. From the foregoing considerations, the SELECT statement which is assigned to a SYSIN data set must be ASSIGNed to a device independent DCB (i.e., UTILITY).

5. SYSOUT goes to disk and is not entered into the writer's queue until the job is terminated. Those data sets assigned to SYSOUT must also have device independent DCB's. Also, remember that large volume outputs could exceed direct access space availability if assigned to SYSOUT.

OS/360 COBOL E FIXES FOR RELEASE 10 AND 11

PTF 8605 fixes a "GO TO DEPENDING ON" problem in level 10 and is included in level 11. PTF 9115 fixes a "PERFORM" problem in both 10 and 11, PTF 9959 will fix a "multiple WRITE in one paragraph" problem in both 10 and 11. Information in PTF's can be obtained through the servicing Field Engineering Branch Office.

OS/360 UTILITIES - DD STATEMENT REQUIREMENTS

The following concerns the DD statement requirements for OS/360 Release 11 Utilities.

The system utilities IEHLIST, IEHMOVE and IEHPROGM require DD statements to allocate volumes rather than data sets. Therefore, the normal rules for DD statements must be modified.

The unit and volume parameters are necessary to allocate volumes. The disposition must be stated as KEEP, although volume disposition is meaningless, because at the end of the system utilities, the scheduler must handle disposition. Therefore, the last data set referenced on an allocated volume is disposed by the scheduler using the disposition for that volume. KEEP must be specified or the last used data set will be deleted since the default option is delete. (For example, if in the IEHLIST program, the last request was to list the PDS SYS1.LINKLIB, SYS1.LINKLIB would be the data set to which the volumes disposition would be applied.)

Therefore, in allocating volumes, specify a disposition of KEEP (or simply OLD which implies KEEP), for a specific volume reference and unit. Since no DSNAME parameter is coded, the DD statement causes allocation of a volume regardless of its mount and allocation characteristics. (A specific volume request for a temporary data set can be satisfied by any possible volume state.)

However, there are times when a non-specific volume request in a DD statement would be necessary. This occurs, for example, where a system utility will want to access more volumes than the number of devices available - an IEHLIST program is asked to list the VTOC of many volumes, but only one device is available. (In this case, the DD statement is used to allocate a device rather than a volume.)

The user must code unit, a disposition of (NEW, KEEP) and a volume parameter value of PRIVATE (with no specific volume reference). This DD statement (without the PRIVATE parameter) represents a non-specific volume re-

quest for a temporary data set (no DSNAME appears), a request which can be satisfied by a public or storage volume, either permanently resident or reserved, or a removable public volume. (A private volume can only be allocated with a specific volume reference.) The use of PRIVATE ensures that a removable public volume will be allocated. (PRIVATE modifies the DD request so that even though no DSNAME parameter is present, the request is for a non-temporary data set.) The removable public volume is demounted (by the scheduler if no DEFER, by OPEN if DEFER is coded in the UNIT parameter) and a private volume mounted. The use of DEFER will save a mount and demount message.

Once the volume is considered private, the system utilities can have it demounted and the desired volume specified in the utility control statement be mounted. Only private removable volumes can be demounted by system utilities.

OS/360 SORT - FILE SIZE ESTIMATE IS NECESSARY

The OS/360 Sort runs at reduced speed when an estimate of file size is omitted. This is due to the sorting method which is chosen by default. A report of a Release 10 sort has been received where this omission cut the speed and the capacity of the sort by one half. The addition of an estimated size in the SORT card restored the capacity and speed.

OS/360 MASTER SRL INDEX

In response to many customer and branch office requests, a Master Cross Reference Index has been prepared for the OS/360 SRL series. The publication is based on the indexes of 39 separate SRL publications. It is entitled "IBM System/360 Operating System: Master Index" (C28-6644). The alphabetic quick-reference index shows the publication title and form number for each index entry.

The initial version of the index is current through OS/360 Release 11. It is available for ordering by IBM Branch Offices from the IBM Distribution Center, Mechanicsburg.

DOS/360 FILES SHARING SAME DASD

Label and extent information is checked by DOS/360 in such a way that the following points should be noted:

1. If two files are physically located on the same DASD device, they should be assigned by means of two separate symbolic names, i.e., SYSxxx & SYSyyy. This is to allow proper extent checking for each file.

2. If two files, even if they are only temporary work files, are assigned to the same DASD device, each MUST have a unique label, i.e., a unique DLAB statement. While some previous release levels of DOS/360 would accept (and work properly with) identical labels for two or more data (or work) files, Version II (Release Level 6 and later) will not; the VTOC of the DASD device will be improperly constructed and job cancellation will result at some later point during execution of the program.

DOS/360 HANDLING MULTI-VOLUME DISK FILES

Sequential disk files under DOS/360 which extend beyond the limits of one pack create a unique situation which requires special attention to operating procedures and the use of proper Job Control statements. Examples are included in this discussion to show how to handle a multi-volume file on one disk drive, by changing packs when one becomes full, or how to handle the same situation when the second disk pack is on a separate drive. Extension of the logic to three or more packs should be obvious from the discussion and examples.

Unlike a direct-access or index-sequential file which requires all volumes to be on line when the file is opened, sequential files, when opened, check only the first extent given by Job Control and then cause additional extents to be opened only when the data within the first extent is exhausted by the problem program. The following is a discussion of what happens while reading a multi-volume file from one disk drive:

1. As the program is being read by Job Control, the information from each VOL, DLAB, XTENT set is combined and written onto the special label information cylinder of the

SYSRES packs (the cylinder immediately following the last SYSRES library). The record for each set contains a key which was the filename specified in the VOL card and which matches the symbolic DTF name in the program.

2. When the file is opened, the filename which was generated as a constant by the DTF name in the program is used to search the special label cylinder of the SYSRES pack for a matching key. When a hit is made, the record portion is extracted and the 44 byte file identification is extracted from the record provided by the earlier DLAB/XTENT information.

3. The 44 byte file identification is then used to match with the format 1 label on the VTOC of the disk drive which was assigned to the SYSXXX unit for that file. Files with multiple extents may have a format 3 label chained to the format 1 label to provide room for additional extents.

4. The volume serial numbers on each volume must be different to identify a new volume. The file serial number is constructed by DOS/360 from the volume serial number of the first volume of the file.

5. When a sequential data set is exhausted on one volume by the program, the next XTENT is automatically opened by logical IOCS. Then the volume serial number is checked against the VOL 1 label of the pack on the drive specified by that SYSXXX assignment. At this point, if the proper second pack is on line, processing continues; and if the proper volume is not available, a message is issued to the operator and processing resumes when the proper pack is mounted.

The examples shown in Exhibit 1 should illustrate the above logic, and intuition should show how the same logic holds when the file is an output file and the multiple packs are on the same drive, or on a separate drive.

DOS/360 HANDLING MULTI-VOLUME DISK FILES

```
// JOB SAMPLE1
* TO READ A MULTI-VOLUME SEQUENTIAL DATA SET FROM ONE DRIVE
// ASSGN SYS0 04,X'19 1'
// VOL S YS 004,DTFNAME
// DLAB  'ALPHAMERIC FI LE NAME                          1222222' X
                  000 1,67001,67001,'DOS                ',SD
// XTENT  1  ,0,00 000 1000,0001 00990,'222222',SY S004
// XTENT  1  ,1,00 007 8000, 0001 23009,'333333',SY S004
// XTENT  1  ,2,00 005 0000,0001 97009,'444444',SY S004
// EXEC SAMPLE1
/&


// JOB SAMPLE2
* TO READA MULTI-VOLUME SEQUENTIAL DATA SET FROM MULTIPLE
* DRIVES
// ASSGN SYS004,X'191'
// ASSGN SYS005,X'190'
// ASSGN SYS006,X'194'
// VOL SYS004,DTFNAME
// DLAB  ' ALPHAMERIC FILE NAME                          '222222' X
                  0001, 61001, 67001, 'DOS              ',SD
// XTENT 1,0,000001000,000199009,'222222',SYS004
// XTENT 1,1,000078000,000123009,'333333',SYS005
// XTENT 1,2,000050000,000197009,'444444',SYS006
// EXEC SAMPLE2
/&
```

Exhibit 1

DOS/360 BASIC FORTRAN TECHNIQUE TO PROCESS 11 RECORDS PER TRACK

1. Description

APAR's have been submitted on the subject of DOS/360 FORTRAN's inability to read more than ten sequential disk records per track. These APAR's cover the specific instance of reading 256 character records written by DOS/360 components, other than FORTRAN, which determine records-per-track from device constants and hence write 11 records per 2311 track.

There is no technical difficulty in altering FORTRAN to read and write 11 records per track; extra code must be added to accommodate existing files written by FORTRAN at ten records per track.

A final fact must be considered: there is no immediate necessity to remove the records-per-track incompatibility, since an existing DOS/360 facility (the direct-access READ statement) may be used to process sequential files written at 11 records per track.

2. Bypass Technique

An immediate solution of the problem is available as follows:

a. For each file which was written by a component other than FORTRAN to be processed by FORTRAN, a DEFINE FILE statement must be added to the source program. The statement, for 256-character records read under FORMAT control, should be written as follows:

DEFINE FILE n(maxrec, 256, E, asvar)

where n is the FORTRAN unit number (DSRN) maxrec is equal to, or greater than, the number of records in the file, and asvar is an integer variable name. This variable will become the "associated variable" for this file, and should therefore be unique in the program.

b. Each READ statement for such a file must be modified by the addition of the associated variable name as follows:

present: READ(n, f) list
becomes: READ(n'asvar, f) list

where n and asvar are as defined above, f is the statement number of a FORMAT statement, and list is the usual I/O list.

c. The associated variable for each file must be initialized to point to the first record of the file by the addition of either of the following statements preceding the first READ for the file:

```
asvar=1
FIND(n, 1)
```

The second form is to be preferred.

d. Since the language specifications prohibit the use of REWIND, BACKSPACE, and END FILE statements for direct-access files, they should be converted to appropriate redefinitions of the associated variable; i.e. BACKSPACE is converted to a reduction by 1 of the value of the associated variable; and REWIND to either of the initializing forms in item 2c above. END FILE is meaningless for input files and may be eliminated if it appears. The present implementation does perform the functions described above for BACKSPACE and REWIND and ignores ENDFILE, but the use of such a non-standard capability is not recommended.

e. Certain restrictions will apply to this technique:

1) A direct-access WRITE may not be used to extend the file unless additional records have been pre-formatted. (Direct-access WRITE's may be used freely within the existing file area to update records, with the side benefit of not destroying the succeeding records.)

2) Extra core will be required at object-time for IJTADIR, the direct-access I/O interface. The present size of this routine is about 1000 bytes.

f. The technique may be extended to read records of any fixed length from 1 to 1726 bytes by changing the constant (256) in the DEFINE FILE statement. This may permit users to specify record

lengths less than 256 bytes for some applications, and thus improve disk space utilization.

DOS/TOS/360 OPERATOR-TO-SYSTEM COMMUNICATIONS

Exhibit 1 is a listing of the various operator-to-system communications commands available under TOS and DOS/360. The four columns represent:

Column 1	BG	Job-Control commands.
Column 2	AR	Attention-Routine commands.
Column 3	Fn	Foreground-Initiator commands.
Column 4	BG	Job-Control statements.

1. Column 1 BG

   Job-Control commands. WITHOUT slashes in the first two columns. They may be entered from either SYSLOG or SYSRDR. Job-Control is a BACKGROUND program, and as such, is available immediately after IPL; it is also called in between background job steps. It is terminated after the execute statement is read. Job-Control control of the 1052 may be suspended via the BG command STOP, may later be re-instated via the AR command START BG.

2. Column 2 AR

   Attention-Routine commands. These MUST be entered from SYSLOG. The AR is called into the supervisor logical transient area, if that area is not active with another B-type transient, when the 1052 REQUEST key is pressed. It is terminated by the START, CANCEL, and MSG commands.

3. Column 3 Fn

   Foreground-Initiation commands. These may be entered from SYSLOG or from a card reader whose address is supplied by the Fn command READ x'cuu'. This READ command MUST be entered from SYSLOG in order to assign the reader, which can then read the other Fn commands. (NOTE: In order to use SYSRDR for Fn commands, SYSRDR may be un-assigned by a Job-Control statement, then the reader assigned for foreground initiation via the Fn READ command.) The Foreground Initiator is a 2K self-relocatable program called into core via the AR command START Fn. It is terminated by the Fn commands CANCEL and EXEC.

4. Column 4 BG

   Job-Control statements, with // in columns 1 and 2. These should be entered from SYSRDR, although some will be accepted from SYSLOG at certain times (e.g., LISTIO entered from SYSLOG or SYSRDR will print on SYSLOG, while // LISTIO entered from SYSLOG or SYSRDR will print on SYSLST). These statements will usually be part of the normal job stream. This job-control is the same as that used with the section in column 1 (above) and is started and terminated in the same way as above.

TOS/360 SYSGEN SRL CORRECTION

The SRL IBM S/360 TOS System Generation and Maintenance (C24-5015-3), erroneously shows, on page 33, the control card to link edit the Tape Compare Utility program as INCLUDE IJJCPIN. The card must be punched INCLUDE IJJCP1N (a 1 instead of an I).

S/360-20 TPS LOADING PROGRAMS FROM CARDS

This alteration to Job Control of the S/360-20 Tape Programming System provides additional functional capabilities. It gives the user the ability to load object programs from cards when using the tape resident system. In order to make use of this option, SYSIPT must be assigned to the primary card reader of the system, and the control card //bEXECbLOADER, C submitted to Job Control. This card must be in the exact format specified. No additional spaces between fields may be taken. The LOADER, C option processes TXT, XFR, END, and REP cards. In addition, the patch contains the normal Job Control functions of PAUSE (only if called for) and skip to 1 after logging of Job Control cards. If logging has been discontinued before the //bEXECbLOADER, C card is encountered, the PAUSE will be displayed on the CPU but not the printer, and the skip to 1 will not be taken. This follows the normal Job Control procedure.

DOS/TOS/360 OPERATOR-TO-SYSTEM COMMUNICATIONS

JOB CONTROL COMMANDS

CAN BE ISSUED BETWEEN BG JOB STEPS. JOB-CONTROL PROG IS A BG PROG.

```
ADD  X'cuu'[(K)],devicetype[,X'ss']
DEL  X'cuu'
SET  DATE=mm/dd/yy CLOCK=hh/mm/ss
SET  [DATE=mm/dd/yy][,CLOCK=hh/mm/ss]
     [,UPSI=nnnnnnnn][,LINECT=nnn]
     [,RCLST=nnn][,RCPCH=nnn]

ASSGN SYSxxx,X'cuu'[,X'ss'][,TEMP]
      "       "    [,ALT]  "
CLOSE SYSxxx[,X'cuu'[,X'ss']]
      "      [,UA]
      "      [,IGN]
      "      [,ALT]
DVCDN X'cuu'
DVCUP X'cuu'
MTC   opcode,SYSxxx[,nn]
      "      ,X'cuu'[,nn]
RESET SYS
      PROG
      ALL
      SYSxxx
STOP                    (BACKGROUND)
LISTIO SYS              (ON SYSLOG)
       PROG
       F1
       F2
       ALL
       SYSxxx
       UNITS
       DOWN
       UA
       X'cuu'
LOG
NOLOG
CANCEL                  (BACKGROUND)

PAUSE
MAP
ALLOC  Fn=nK[,Fn=nK]
UCS    SYSxxx,phasename,etc.
HOLD   Fn
RELSE  Fn
UNA    Fn
```

ATTN. COMMANDS

CAN BE ISSUED ANYTIME PRESS REQUEST-1052

```
LOG
NOLOG
CANCEL BG
       F1
       F2
PAUSE
MAP
ALLOC  Fn=nK,etc.

START  BG
       F1
       F2
MSG    F1
       F2
TIMER  BG
       F1
       F2
```

FOUNDATION

CAN BE ISSUED AFTER TYPING START F4-1052

```
LOG
NOLOG
CANCEL (INITIATOR)

HOLD   Fn
RELSE  Fn
UNA    Fn

READ X'cuu'

VOL    SYSxxx,filename
DLAB   'lbl fields 1-3',etc.
XTENT  type,seq,etc.
TPLAB  'lbl fields 3-10'
TPLAB  'lbl fields 3-13'
EXEC   progname
```

JOB CONTROL STATEMENTS

CAN BE ISSUED BETWEEN BG JOB STEPS JOB-CONTROL PROGRAM IS A BG PROGRAM.

```
// JOB    jobname
// DATE   mm/dd/yy
// ASSGN  SYSxxx,X'cuu'[,X'ss']
       "       "    [,ALT]
// MTC    opcode,SYSxxx[,nn]
// RESET  SYS
          PROG
          ALL
          SYSxxx
// LISTIO SYS              (ON SYSLST)
          PROG
          F1
          F2
          ALL
          SYSxxx
          UNITS
          DOWN
          UA
          X'cuu'

// PAUSE

// UPSI   nnnnnnnn

// LBLTYP TAPE[(nn)]
// LBLTYP NSD(nn)
// NMTLB  n

// RSTRT  SYSxxx,nnnn
// OPTION option1,option2,etc.
// VOL    SYSxxx,filename
// DLAB   'lbl fields 1-3',etc.
// XTENT  type,seq,etc.
// TPLAB  'lbl fields 3-10'
// TPLAB  'lbl fields 3-13'
// EXEC   progname
/*
/&
*
```

Exhibit 1

The user can assemble his own patch from the enclosed specifications. He should remove the END card of Job Control and replace it with the assembled patch cards. The patch contains a correct END card. Follow the normal CMAINT procedure to place the patched Job Control on SYSRES.

The Table of Defined Values is shown in Exhibit 1 and the source and object code is shown in Exhibit 2.

This alteration has been tested only by the author for Version 1 Modification Level 1 of Job Control. It is a field contribution and has not been submitted to any formal IBM test. Alteration of cards 01-24 may be necessary to make the patch function on any other version or level.

S/360-20 TPS LOADING PROGRAMS FROM CARDS

TABLE OF DEFINED SYMBOLS

SYMBOL	LEN	VALUE	TYPE
BADR	2	122C	X
DIFF	2	11CE	X
END	4	121C	I
ENDA	1	10CA	
ERDR	4	122E	I
G003	1	0000	*
HALT	4	1234	I
H2	2	11D0	X
H5	2	11D2	X
LB01	4	1166	I
LB02	4	115E	I
LB03	4	1176	I
LB04	4	115A	I
LB05	6	1186	I
LB06	4	11E2	I
LOC	2	0E12	Y
NEXT	6	124C	I
PH01	4	114C	I
PH02	4	11A0	I
READ	80	1270	C
REP	6	123A	I
TRAN	3	11D4	X
XFR	4	1222	I

Exhibit 1

```
LOCATN  OBJECT CODE   ADD1 ADD2 STMT     SOURCE STATEMENT

                                0001        AOPTN  NOESD,NORLD
 0000                           0002 G003    START  0
 10CA                           0003 ENDA    EQU    G003+X'10CA'          JOB CONTROL START ADDRESS
 0000                           0004         USING  *,0
 1000                           0005         USING    *+4096,1
 0E12                           0006         ORG    G003+X'0E12'          ORG TO BRANCH ADDRESS
 0E12 114C                      0007 LOC     DC     Y(PH01)               SET BRANCH IN JOB CONTROL
 114C                           0008         ORG    G003+X'114C'          ORG TO END OF JOB CONTROL
 114C 954E 0E17       0E17      0009 PH01    CLI    X'0E17',X'4E'         IS HALT 0B4E
 1150 4780 1166       1166      0010         BC     8,LB01                YES - CONTINUE
 1154 D200 1161 F003  1161 0003 0011         MVC    LB02+3(1),3(15)       NO-SET HALT
 115A 4DF0 1024       0024      0012 LB04    BAS    15,X'024'(0,1)        BRANCH TO JOB CONTROL ROUT
 115E 9900 0B00       0B00      0013 LB02    HPR    X'B00',0              DISPLAY HALT
 1162 47F0 0892       0892      0014         BC     15,X'0892'            RETURN TO JOB CONTROL
 1166 95C3 0741       0741      0015 LB01    CLI    X'0741',C'C'          IS THIS LOADER,C
 116A 4780 1176       1176      0016         BC     8,LB03                YES - GO TO LOAD ROUTINE
 116E 924E 1161       1161      0017         MVI    LB02+3,X'4E'          NO - SET HALT
 1172 47F0 115A       115A      0018         BC     15,LB04               BRANCH TO HALT HANDLER
 1176 9104 B025       0025      0019 LB03    TM     X'025'(X'B'),X'04'    IS PAUSE BIT ON
 117A 4780 1186       1186      0020         BC     8,LB05                NO - BYPASS PAUSE
 117E 4DF0 1024       0024      0021         BAS    15,X'024'(0,1)        YES - GO TO PAUSE HANDLER
 1182 9900 0B40       0B40      0022         HPR    X'B40',0              DISPLAY PAUSE
 1186 D205 104C 0ADE  004C 0ADE 0023 LB05    MVC    X'04C'(6,1),X'ADE'(0) SET BRANCHES IN PRINT ROUT
 118C 4DF0 103C       003C      0024         BAS    15,X'03C'(0,1)        BRANCH TO SKIP HANDLER
 1190 D2DF 0680 11A0  0680 11A0 0025         MVC    X'0680'(224),PH02     MOVE LOADER TO 680
 1196 D201 00C8 085C  00C8 085C 0026         MVC    X'00C8'(2),X'085C'    STORE FETCH ADDRESS
 119C 47F0 0680       0680      0027         BC     15,X'0680'            BRANCH TO LOADER
 11A0 8100 0684       0684      0028 PH02    SPSW   X'0684'               BRANCH AND DISABLE INTERPT
 11A4 0000068E                  0029         DC     X'0000068E'           PSW
 11A8 0A0B0C0D0E0F              0030         DC     X'0A0B0C0D0E0F'       TRANSLATE TABLE
 11AE 0D80                      0031         BASR   8,0                   SET BASE REG
                                0032         DROP   0                     DROP BASE 0
                                0033         DROP   1                     DROP BASE 1
 11B0                           0034         USING  *,8                   ESTABLISH NEW BASE
 11B0 D012 80C0 0050  1270 0050 0035         XIO    READ(X'12'),80        READ CARD
 11B6 0748                      0036         BCR    4,8                   WAIT IF WORKING
 11B8 4710 8084       1234      0037         BC     1,HALT                BRANCH IF READER NOT READY
 11BC 9A10 800C       11BC      0038         TIOB   *,X'10'               WAIT IF READER BUSY
 11C0 9A11 807E       122E      0039         TIOB   EROR,X'11'            BRANCH IF READ ERROR
 11C4 9502 80C0       1270      0040         CLI    READ,X'02'            IS THIS A LOADER CARD
 11C8 0778                      0041         BCR    7,8                   NO - READ ANOTHER
 11CA 47F0 8032       11E2      0042         BC     15,LB06               BRANCH TO CONTINUE
 11CE D1FF                      0043 D1FF    DC     X'D1FF'               BASE TO DEVELOP MOVE INSTR
```

Exhibit 2

```
11D0 0002                                0044 H2      DC    X'0002'                    HALFWORD 2
11D2 0005                                0045 H5      DC    X'0005'                    HALFWORD 5
11D4 000000                              0046 TRAN    DC    X'000000'                  TRANSLATE AREA
11D7 0001020304050607                    0047         DC    X'000102030405060708090'   TRANSLATE TABLE
11E2 9200 80C0       1270                0048 LB06    MVI   READ,X'00'                 RESET LOADER ID
11E6 95C5 80C2       1272                0049         CLI   READ+2,C'E'                IS THIS A REP CARD
11EA 4780 808A       123A                0050         BC    8,REP                      YES - GO TO REP ROUTINE
11EE 48E0 80C6       1276                0051         LH    14,READ+6                  LOAD BASE ADDRESS
11F2 95D5 80C2       1272                0052         CLI   READ+2,C'N'                IS THIS AN END CARD
11F6 4780 806C       121C                0053         BC    8,END                      YES - GO TO ENDCARD ROUT
11FA 95C6 80C2       1272                0054         CLI   READ+2,C'F'                IS THIS XFR CARD
11FE 4780 8072       1222                0055         BC    8,XFR                      YES - GO TO XFR ROUTINE
1202 95E7 80C2       1272                0056         CLI   READ+2,C'X'                IS THIS TXT CARD

LOCATN  OBJECT CODE   ADD1 ADD2 STMT      SOURCE STATEMENT

1206 0778                                0057         BCR   7,8                        NO - READ ANOTHER
1208 48F0 80CA       127A                0058         LH    15,READ+10                 LOAD NO OF BYTES TO MOVE
120C 4AF0 801E       11CE                0059         AH    15,D1FF                    DEVELOP MOVE INSTRUCTION
1210 40F0 8064       1214                0060         STH   15,*+4                     STORE MOVE INSTRUCTION
1214 D200 E000 80D0 0000 1280            0061         MVC   0(1,14),READ+16            MOVE DATA TO CORE
121A 07F8                                0062         BCR   15,8                       READ ANOTHER
121C 9540 80C6       1276                0063 END     CLI   READ+6,X'40'               IS BRANCH ADDRESS BLANK
1220 0788                                0064         BCR   8,8                        YES - READ ANOTHER
1222 40E0 807C       122C                0065 XFR     STH   14,BADR                    SET BRANCH ADDRESS
1226 8100 807A       122A                0066         SPSW  *+4                        ENABLE INTERRUPT AND BRANCH
122A 0100                                0067         DC    X'0100'
122C 0000                                0068 BADR    DC    X'0000'                    PSW
122E 9900 01D1       01D1                0069 EROR    HPR   X'1D1',0                   ERROR HALT
1232 07F8                                0070         BCR   15,8                       TRY AGAIN
1234 9900 01D0       01D0                0071 HALT    HPR   X'1D0',0                   NOT READY HALT
1238 07F8                                0072         BCR   15,8                       TRY AGAIN
123A DC03 80C8 05C7 1278 05C7            0073 REP     TR    READ+8(4),X'05C7'          TRANS LOAD ADDRESS
1240 F224 8024 80C8 11D4 1278            0074         PACK   TRAN,READ+8(5)            PUT IN HEX
1246 48E0 8024       11D4                0075         LH    14,TRAN                    SET LOAD ADDRESS IN REG 14
124A 1BFF                                0076         SR    15,15                      CLEAR BASE REG
124C DC03 F760 05C7 0760 05C7            0077 NEXT    TR    X'760'(4,15),X'05C7'       TRANSLATE DATA
1252 F224 8024 F760 11D4 0760            0078         PACK   TRAN,X'760'(5,15)         PUT IN HEX
1258 D201 E000 8024 0000 11D4            0079         MVC   0(2,14),TRAN               MOVE DATA TO CORE
125E 956B F764       0764                0080         CLI   X'764'(15),X'6B'           IS NEXT CHAR COMMA
1262 0778                                0081         BCR   7,8                        NO - READ ANOTHER
1264 4AE0 8020       11D0                0082         AH    14,H2                      INCREMENT LOAD ADDRESS
1268 4AF0 8022       11D2                0083         AH    15,H5                      INCREMENT CARD ADDRESS
126C 47F0 809C       124C                0084         BC    15,NEXT                    TRANSLATE NEXT TWO BYTES
1270                                     0085 READ    DS    CL80                       CARD READ AREA
10CA                                     0086       END     ENDA                       BRANCH TO JOB CONTROL START
```

Exhibit 2 (continued)

S/360-20 DISK RPG POINTERS

1. Figure 114 in the S/360-20 DPS/TPS RPG SRL (C24-9001-2) should show use of a record length of 8. The block length is specified by the use in the OUTFIL sort control card.

2. TNL N33-9007 to RPG SRL C24-9001-2 describes the coding for indexed sequential load and add.

3. Sequential processing of multiple files with no matching record fields is supported.

4. RPG user programs not using tapes can overlay the tape error routines and error statistics in the monitor based on the load point specified in the RPG control card. However, if a subsequent program requires these routines, the full monitor must be restored by another IPL.

5. Random retrieval of indexed sequential file does not seek the cylinder index when the previous record is on same cylinder.

S/360-20 BIBLIOGRAPHY CORRECTION

An excerpt for PL/I appeared erroneously, on page 8, in the latest Model 20 Bibliography (A26-3565-2). It is an excerpt for DOS/TOS PL/I. There is no work being done on PL/I for S/360-20, nor is any presently planned. The erroneous entry should be in the Bibliography.

S/360-30 TWO MICROSECOND PROCESSOR TYPE I SUPPORT

Current IBM-supplied Type I programs announced for S/360 Model 30 support the 2-microsecond processor, except where user-supplied coding conflicts with requirements of time-dependent devices and where user processing associated with process control and communications devices requires a higher-speed processor.

S/360 RPG CODING FOR COMPATIBILITY

System/360 Report Program Generator is implemented on all models of IBM's S/360. Every effort was made to make S/360 RPG a compatible source programming language on S/360. Most of the differences are minor and are given here along with some guidelines to aid the programmer to code in RPG so that his programs will run with a minimum of changes on all versions of the RPG generator program. The different versions of RPG covered are shown below. RPG for the Basic Programming System (BPS) is not included since this version of RPG has been largely replaced by one of the other versions shown below.

Programming System	Minimum Configuration	Abbreviation	S/360 Model
Model 20 RPG (Card)	4K	Model 20 Card	20
Model 20 Tape Programming System	8K	TPS	20
Model 20 Disk Programming System	12K	DPS	20
IBM Basic Operating System	8K	BOS	30 and up
IBM Tape Operating System	16K	TOS	30 and up
IBM Disk Operating System	16K	DOS	30 and up
IBM Operating System/360	64K	OS	30 and up

In the following description of the differences, the versions of RPG are referred to by the abbreviations shown above.

1. General

   a. Sequence of source specifications for program generation for Model 20 Card RPG is: Header, File Description, File Extension, Input, Output, /* card, (tables and one /* card if tables used), data, /* card.

      Sequence for all other versions of RPG is: Header, File Description, File Extension, Input, Calculation, Output, /* card, data, /* card.

b. For Model 20 Card RPG, table-input records are included with the source deck and included in the generated object program.

For all other versions of RPG, table-input records are omitted during program generation and read into core at the beginning of execution of the object program.

2. File Description Specifications

a. File Designation (position 16) is not required with Model 20 Card RPG. It is required with all other versions of RPG.

b. Block Length and Record Length (positions 20 to 27): With TPS and DPS four must be added to the record length of variable-length tape records, and eight must be added to length of largest variable-length tape record for block length of variable-length tape records.

DPS will not handle variable-length disk records. For BOS, TOS, DOS and OS, enter the exact record and block lengths for variable-length records.

c. Table-input files are not described on the File Description Specifications with Model 20 Card RPG, but they must be for all other versions of RPG.

3. File Extension Specifications

a. Table Name (positions 27 to 32, 46 to 51): Length may be four to six characters with Model 20 Card RPG, TPS and DPS. Length must be six characters with BOS, TOS, DOS and OS.

b. Packed (positions 43, 55): Must be blank for Model 20 Card RPG. May contain blank or "P" for all other versions of RPG, indicating that table-input records are in packed-decimal format.

4. Input Specifications

Control Level (positions 59 to 60): Must be specified in ascending sequence (L1, L2, L3... L9) for Model 20 Card RPG. All other versions of RPG allow specifications in any sequence.

5. Calculation Specifications

a. The RLABL operation may specify a label of one to four characters with Model 20 Card RPG. All other versions of RPG permit a length of one to six characters.

b. The ULABL operation is not permitted with Model 20 Card RPG, TPS or DPS. It is permitted with BOS, TOS, DOS and OS.

c. Two Halt Indicators (H1 and H2) are provided with Model 20 Card RPG, TPS and DPS. BOS, TOS, DOS and OS provide nine Halt Indicators (H1, H2, H3... H9).

d. Indicators entered for the Move Remainder operation (MVR) must be identical to the preceding Divide operation (DIV) with Model 20 Card RPG, TPS and DPS. They are ignored for BOS, TOS, DOS and OS.

e. An RLABL operation and a TAG operation do not permit entries in Indicators (positions 9 to 17) with Model 20 Card RPG, TPS and DPS. All other versions of RPG ignore entries in Indicators for these operations.

6. Output-Format Specifications

a. IBM 2560 MFCM operations (stacker selection to stackers 3, 4 and 5 and document printing) are supported by Model 20 Card RPG, TPS and DPS only, since the MFCM is available only on the Model 20.

b. Model 20 Card RPG has one automatic page numbering counter, PAGE. TPS and DPS have two page counters, PAGE and PAGE1. BOS has three page counters, PAGE, PAGE1 AND PAGE2. TOS, DOS and OS have eight page numbering counters, PAGE, PAGE1, PAGE2, PAGE3 ...PAGE7.

c. Space Before and Space After (positions 17, 18): If left blank with Model 20 Card RPG, causes no spaces to be taken. TPS and DPS do not permit both of these entries to be blank on one line. If left blank with BOS, TOS, DOS and OS, a single space after printing will result.

d. Edit words must be the same size as the

data fields being edited with Model 20 Card RPG, TPS and DPS. Edit words can be equal to, or larger than, the data field being edited with BOS, TOS, DOS and OS.

From the above list, some guidelines for coding the RPG program can be generated and used to provide maximum compatibility with other versions of S/360 RPG. These guidelines include:

1. Always enter a specification in File Designation (position 16) on the File Description specifications.

2. Always use six-character table names.

3. Always specify the Control Level entries on the Input specifications in ascending sequence.

4. With RLABL operations, keep the length of the label specified within one to four characters.

5. Use Halt Indicators H1 and H2 whenever possible, rather than using H3, H4...H9. Consider whether stacker selection or error message would be better than the use of Halt Indicators.

6. Always use the same indicators to condition the Move Remainder operation (MVR) that are used to condition the Divide operation (DIV) that precedes it.

7. Never put entries in Indicators (positions 9 to 17) with the RLABL and TAG operations.

8. Always place an entry in Space After (position 18) on the Output Specifications. For space supression, enter a zero. For single, double or triple spacing enter one, two or three.

9. Always set up the edit word to be the same size as the data field being edited.

10. Avoid using special features on a particular system configuration if you plan to run the program on different systems or models of S/360 which do not have the feature/s.

NEW TYPE III PROGRAMS *

The following are the abstracts of programs which have been recently made available from the Type III library.

The program, along with its complete abstract, will be incorporated in subsequent issues of the Catalog of Programs.

Programs may be obtained by submitting a properly completed "General Program Request Card" (Form Number 120-1145-1) to the Program Information Department, 40 Saw Mill River Road, Hawthorne, New York, 10532.

These programs and their related documentation are distributed by IBM in the author's original form and have not been subjected to any formal testing.

Any discussion of Type III Programs must emphasize the following points:

1. Type III programs are not part of the IBM product line, as are Programming Systems (Type I) or Application Programs (Type II).
2. Type III programs have not been subjected to any formal product test.
3. Recipients of Type III programs are expected to make the final evaluation as to the usefulness of the programs in their own environment.
4. There is no committed maintenance for Type III programs. However, any changes the author chooses to make will be announced in subsequent issues of the Catalog of Programs.

* NOTE: THE CUSTOMER MUST BE INFORMED THAT THE ABOVE APPLIES TO ANY OF THE FOLLOWING TYPE III ABSTRACTS FURNISHED TO HIM.

S/360

DISK AND CORE REQUIREMENT ANALYZER FOR THE S/360 BILL OF MATERIAL PROCESSOR. This consists of two programs written in FORTRAN and run on either an IBM 1130 or or S/360 computer. The first program analyzes master file requirements, and the second program analyzes chain file requirements. These programs follow closely the analysis presented by the flow charts in Appendix E of the S/360 Bill of Material Processor Programmer's Manual (H20-0246). These programs provide no single, definite solution, but instead a number of possible solutions which are presented in tabular form. The IBM representative and customer can analyze and weigh the advantages of one solution over another and arrive at the optimal solution tailored to the customer's needs and hardware configuration. These programs save many hours of manual calculation which normally just would not be performed to arrive at the optimal solution for a particular installation.

System Configuration: For S/360: 16K Processing Unit (Model D) with Floating Point Option, Card Reader (any S/360 model), Printer (any S/360 model with at least 120 print positions). For the 1130: 8K Processing Unit (Model 1B), 1442 Card Read Punch, 1132 Printer.

Ordering Procedure: Order File Number 360D-12.1.007. The program material can be obtained in card form or on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR). The requester must indicate whether a 9 track or 7 track DTR is required. If not specified, a 9 track DTR will be provided. No tape submittal is required - the DTR will be provided by the Library.

TAPE FILE CREATION PROGRAM. This program produces S/360 tape files with blocked or unblocked variable or fixed length records from card input. The BPS utility and the Autotest card-to-tape programs are very limited when output records are other than fixed length or have varying formats. This program will write packed, unpacked or binary data on the tape according to field control characters punched in data cards. There is no need for the user to determine the block count and/or record count for variable length records, as in autotest, nor are the blocks limited to 400 characters. Ease of use and flexibility of output are the main criteria for using this program in preference to the current utilities mentioned.

System Configuration: Written in S/360 assembler language, the program requires less than 16K of core (including supervisor and job control) and one 2400 series tape drive. It is extremely valuable in pre-installation testing.

Ordering Procedure: Order File Number 360D-00.3.004. The program material can be obtained in card form or on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR). The requester must indicate whether a 9 track or 7 track DTR is required. If not specified, a 9 track DTR will be provided. No tape submittal is required - the DTR will be provided by the Library.

EASING DOS TPLAB RESTRICTIONS. Current implementation of the //TPLAB card has the requirement that all fields (reel serial, etc.) are to be punched prior to use.

This package consists of two transient routines inserted in the string of routines used to open tape files. These routines cause, in general, only those fields that are punched to be checked while those fields that are blank will be filled with proper data before label checking is performed.

Retention period facilities are provided as well as optional indication on SYSLOG that opening is occurring.

Ordering Procedure: Order File Number 360D-03.4.012. The program material can be obtained in card form or on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR). The requester must indicate whether a 9 track or 7 track DTR is required. If not specified, a 9 track DTR will be provided. No tape submittal is required - the DTR will be provided by the Library.

1130

IDEAL STR COMMUNICATION IN IBM 1130 FORTRAN WITH BCD, EBCDIC AND BINARY I/O. This package of subroutines provides the IBM 1130 FORTRAN user with the required calls to operate the Synchronous Communica-

tion Adapter (#7690) in STR mode. Both BCD and EBCDIC subset data codes are supported for communication with existing STR terminals. Data I/O is in FORTRAN A1 format. No internal restrictions exist as to transmittal record size, number of subtransmittal records, data buffering or terminating group marks (338 words+SCAT1+STRTB+HXCV). New "IDEAL" I/O subroutines are provided to support concurrent 1132 printing with SCA operation (192 words+PRNT2), Column Binary card reading and punching (274 words+CARD1). Modify CNVRT for 256 character conversion (32 words). Display variable in A&Q (14 words). Disk I/O with SCA (18 words). All are written in Assembler language. Six FORTRAN demonstration programs are provided to exercise these routines.

System Configuration: Useful non-disk I/O IDEAL FORTRAN communications programs fit 4K systems. Column binary card I/O may be used in any program. User FORTRAN programs must use one word integer option.

Ordering Procedure: Order File Number 1130-03.0.006. Distribution will be in card form only. If source decks are desired, they must be ordered as optional material.

INVENTORY SIMULATOR - FORTRAN IV. This program, which allows the simulator options to be selected by means of a control card, is a modified version of the Modular Inventory Simulator, PID number 1401-CS-03X or 1620-CS-04X, version 1, modification level 0. The concept of a Beta factor which expresses the relationship of MAD and time has been included in this program.

The use of an inventory simulator has proven to be a valuable technique for defining the objectives of an inventory system and for pretesting the decision rules selected to accomplish these objectives.

System Configuration: This program is written in FORTRAN IV and was compiled and tested on a 8K IBM 1130 using the IBM 1130 Monitor FORTRAN. However, since this program was written in FORTRAN IV, it could be run on any system, i.e. S/360, 1800, etc., which has a FORTRAN IV compiler and enough core.

Ordering Procedure: Order File Number 1130-15.7.001. Distribution will be in card form only.

1401

TALLY ANALYSIS REGISTRATION AND SCHEDULING - 1401. TARS - 1401, written in Autocoder, is designed for an 8K, 1-disk 1401 System for Student Scheduling in secondary schools. The programs produce aids to Master Schedule Building (Tally and Conflict Matrix) and, using a master schedule prepared by the school, process student course requests to create student schedules and an updated master schedule.

This contribution represents a conversion of Program Number 1440-11.3.001* which is a 1440 student scheduling package. With the new program, schools with a 1401 installation will be able to schedule students on their own equipment.

Ordering Procedure: Order File Number 1401-14.0.025. Distribution will be in card form only.

* A new version of the 1440 TARS program is now available from the Type III library. Users of the 1440 TARS program should order a complete new program using File Number 1440-11.3.001.

IBM Installation Newsletter

July 14, 1967

Issue No. 67-14

CONTENTS
(G means Green Section)

Distribution: Branch Offices - DP Management, Salesmen, Systems Engineers, FE Managers. Regions, Districts, Education Centers, Field Systems Centers, Federal Systems Center, FE Area Offices, DPD HQ, FED HQ, WTC.

* Requires Immediate Attention

For IBM Internal Use Only

DPD HQ, White Plains, N. Y.

OS/360 RELEASE 11 - SYSTEMS GENERATION CONSIDERATIONS

SYS1.MACLIB, blocked, requires 460 tracks instead of 410 as indicated in the Storage Estimates manual. This results in a Format 3 DSCB in the VTOC if less than 460 tracks are allocated to SYS1.LIB.

If you are planning to use F level linkage editor, PTF's 8982, 9007 and 1064 must be applied before beginning Stage II. However, there is not enough room in SYS1.LINKLIB in the distributed starter system to accommodate an enlarged SYS1.LINKLIB. The following procedure may be used:

1. Pre-allocate 45 cylinders, 50 directory entries in a data set with a unique name (HOPE), on DLIB01.

2. Move SYS1.LINKLIB, using IEHMOVE utility program, to the new area on DLIB01, using rename to HOPE.

3. Scratch SYS1.LINKLIB, rename HOPE as SYS1.LINKLIB.

4. Apply PTF's.

OS/360 BISAM - DETERMINATION OF CORE SPACE

A routine has been developed to determine dynamically the core space required by the highest level of index, and place this information in the DCB. This routine utilizes the DCB exit at OPEN time to access the routine which will extract the 'NCRHI' value from the format 2 DSCB for the data set. At the time OPEN exits to the DCB modification routine, the following events have taken place:

1. OPEN has constructed a work area for the DCB being OPENED and has placed the work area address in the DCBDEBAD FIELD of the DCB (DCB+44).

2. OPEN has read the FORMAT 1 DSCB and the JFCB into the workspace and has filled in the DCB with FORMAT 1 information.

3. OPEN has read the FORMAT 2 DSCB into an additional work area, but has not filled in the DCB fields. The address of the FORMAT 2 DSCB can be found in the OPEN work area following the FORMAT 1 DSCB at work area + 96.

In the exit routine, the user can find the 'NCRHI' value at (FORMAT 2 DSCB + 64). Using this value, he should allocate space by GETMAIN and fill in the DCBMSHI (DCB+72) and the DCBSMSI (DCB+68) fields. Later in its execution OPEN will read the highest level of INDEX into the allocated space. See Exhibit 1 for the exit routine. Users should be aware that this routine has not been submitted to any formal IBM test.

NOTE: Users DCB must use EXLST parameter and point to address of the exit list. In Exhibit 1, it is EXLST=LIST.

OS/360 AND DOS/360 QTAM

At the present time, QTAM is designed in such a manner that there is no "wrap-around" to reuse space on the disk queue.

The only way to start back at the top of the pack (beginning of the queue) is by re-IPLing.

This may present a severe limitation for a customer who is attempting to run a 24-hour T/P operation, or one who has a high volume of traffic during a shift or day.

The current version of QTAM will go into a loop if the space on the disk queue becomes filled.

OS/360 LINKAGE EDITOR F

This linkage editor is being withdrawn from OS/360 Release 12. An article in Installation Newsletter issue 67-12 (page 5) described the differences between the E (44K) and F linkage editors, plus the procedure which would be followed when eliminating LE-F. Users requiring LE-F to perform the transition process should plan to retain their F editor from an earlier system, or obtain assistance from Field Engineering.

OS/360 BISAM - DETERMINATION OF CORE SPACE

```
1.          DCBD        DSORG=IS
2.  BISAM   CSECT
3.          USING       IHADCB,(X)          X=Register of users choice
4.  OPEN1   OPEN        (NAME OF USERS DCB)
5.          B           (BRANCH AROUND FOLLOWING ROUTINE)
6.          DS          ØF
7.  LIST    DC          X'85'               POINTER IN EXIT LIST
8.          DC          AL3(EXIT)           TO USERS ROUTINE
9.  EXIT    LA          X,(USERSDCB)        X=REGISTER USED IN THE
                                            ABOVE USING STATEMENT
10.         L           8,DCBDEBAD          Load Pointer to open
                                            work area - DCB+44
11.         L           7,96(8)             Extract ADDR of format
                                            2 DSCB from work area
12.         CLI         Ø(7),X'Ø2'          Test for code in first
                                            byte of DSCB
13.         BNE         RETURN              Return if not FORMAT 2
14.         CLI         44(7),C'2'          Double ck-FORMAT 2 IDENT.
15.         BNE         RETURN
16.         LH          Ø,64(7)             'NCRHI' value is at location
                                            64 in DSCB 2
17.         STH         Ø,DCBSMSI           Establish index size in
                                            DCB - DCB+68
18.         GETMAIN     EC,LV=(Ø),A+TEMP    Return code in R15
                                            Ø = space alloc
                                            4 = no space avail
19.         B           *+4(15)             Test return code
20.         B           AROUND              Space was allocated
```

Exhibit 1

OS/360 BISAM-DETERMINATION OF CORE SPACE

```
21. RETURN   BR     14                    Go back to OPEN
22. AROUND   MVC    DCBMSHI+1(3), TEMP+1  (OR DCB + 72)
                                          establish index address
                                          in DCB without destroying
                                          high order byte of the MSHI
                                          word - it contains 'NCP'
23.          RETURN
24. TEMP     DS     F
```

Exhibit 1 (continued)

The most significant difference between the LE-E (44K) and the LE-F was the number of overlay segments which could be handled (63 compared to 255). Release 12 will expand the LE-E (44K) capability to 255, thus removing the major limitation. Consideration is being given to a new improved linkage editor F to replace that withdrawn from Release 12, although plans are not yet firm.

DOS/360 SEQUENTIAL DATA SETS CREATED ON A 1316

DOS/360 sequential data sets created on a 1316 with deleted data sets are incompatible with M44PS. DOS/360 does not update the last Format-1 field in the Format-4 label. Assuming N deleted data sets on the DOS/360 1316, one must allocate N+1 M44PS data sets to this 1316 which will cause the M44PS to update the label.

NOTE: Before using any DOS/360 created 1316 with M44PS, execute the utility map function upon it to update the Format-5 label. The number of deleted data sets will be noted on the map print out.

DOS/360 ASSEMBLER OPTIONS CORRECTION

Issue No. 67-11 contained, on page 5, an article titled "DOS/360 Assembler Options." Paragraph 3 of that article indicated a restriction pertaining to the intermixing of work units which does not exist. The corrected paragraph follows:

> Additional Core (IJQD32). This option requires 14K bytes of main storage and allows SYS001, -002, -003 to reside on disk or tape work units. Additional main storage, if available, will be used to expand certain tables. The work areas and I/O buffers also are expanded. Certain sub-phases which are normally overlays are loaded in together. The maximum core used is somewhat less than 60K.

S/360 - 7070/7074 SIMULATOR 360C-S1753

The performance information contained in SRL C28-6530-1 applies only to the simulation of 5,000 or 10,000 word 7070/7074 programs.

7074 programs that use more than 10,000 words will be executed by the simulator at approximately 80 percent of the 5,000 or 10,000 word performance.

A TNL containing this new performance information will be released.

S/360 MODEL DEPENDENCIES

The following article concerns the compatibility aspects of S/360 and the implications of using S/360 functions that are defined to be unpredictable (see S/360 Principles of Operation, A22-6821, pages 5 and 161-164).

Three types of model dependencies may be distinguished in S/360: model dependencies due to implementation, due to physical capabilities, and due to logical capabilities.

Each model of S/360 is unique in its physical implementation. This uniqueness is reflected in the programming aspects that pertain to the diagnosis and maintenance, such as the execution of the instruction DIAGNOSE, the causes of the machine-check interruption, and the format and amount of the information scanned out upon detection of equipment malfunctioning. This information is essential for maintenance of the machine, and must be made available to those concerned with these functions. This information, including the operation of DIAGNOSE, however, does not concern and must not be used by those who want to benefit from the compatibility of S/360.

The physical capabilities include such aspects as execution times, I/O data rates, and the systems facilities available. These capabilities differ among models and are significant attributes of a system. Information on these aspects is widely disseminated. These capabilities are a measure of what resources are available and how fast a task can be performed.

To properly utilize a system, the program must be aware of the resources, and hence the programmer must have the information. The presence of the required physical facilities, in fact, is one of the constraints on compatibility of S/360 (see page 5 of Principles of Operation, A22-6821). Dependence on a facility, such as the floating-point feature, means that a program will not run on a system without this facility; but as long as the requirements for the systems facilities are satisfied, the program runs on any present or future model regardless of the identity of the CPU.

The above applies also to the information on CPU and I/O timing. This information provides a measure of system performance and is needed both for configuration and for efficient utilization of the system. In using timing information, one must be careful, however, to avoid implicit time dependencies in the program, i. e., the compatibility rules of S/360 hold only when the program is independent of the relation of the instruction execution times and of I/O data rates, access times, and command execution times.

One of the basic and most significant goals of S/360 was that all models perform the logical and arithmetic functions identically, provided the constraints on the required physical facilities and independence of time relations are satisfied. Considering the complexity of the machines and the number of alternatives in which each function could be performed, this goal has been achieved to a very high level. Identical operation, however, could not be justified for a number of functions concerning invalid programs or machine malfunctioning.

These functions are those for which neither frequency of occurrence nor usefulness of results warrant identical action on all models.

Whenever non-identical operation was permitted, the result was said to be unpredictable. All functions that may differ among models are described on pages 161-164 of the S/360 Principles of Operation SRL. These functions include all logical or arithmetic operations where compatibility is not maintained, as well as the implementation-oriented functions (e. g., DIAGNOSE) which may, by some users, be mistakenly assumed to be a user-oriented logical facility.

The intent is that the user must ignore any results that are defined to be unpredictable. Thus, no particular accommodation is required for these functions; the only requirement is that users should not depend on any particular value of a result that is defined to be unpredictable, realizing that the function may differ among models.

If the user understands the reason and the meaning of the model-dependent functions, he should not be concerned with finding out how each machine performs the incompatible functions. Therefore, this information is not made generally available; instead, more emphasis is put on educating the users on the meaning and value of compatible architecture and on ways of benefiting from it.

There will, of course, be exceptions to the type

of user as originally envisioned by Systems Architecture. There will be users who either have specialized applications or want to sacrifice the value of uninhibited future growth for additional function. Such use is equivalent to installation of an RPQ function.

NEW PUBLICATIONS ANNOUNCED BY PRL'S

The weekly PRL's (Publications Release Letters) are used to insure that all Salesmen and Systems Engineers are aware of new or revised Marketing Publications. Normally, each issue of the Newsletter will contain information from two PRL's, one following the other. The information will be placed in the Newsletter in its original form with no rearrangement of form numbers or titles. It is not intended to replace existing information and distribution sources. You should be certain that you are aware of these sources.

INFORMATION ABOUT THE NEWSLETTER

EVERY SALESMAN AND SYSTEMS ENGINEER SHOULD RECEIVE ONE COPY OF THE NEWSLETTER. Each FE Branch Manager should receive five copies for distribution to Customer Engineers.

The IBM Installation Newsletter is distributed only to IBM locations and is not available to customers. The Newsletter is separated into two parts. The first part contains information which is intended only for IBM personnel and the pages are labeled For IBM Internal Use Only. THESE PAGES, OR REPRODUCTIONS OF THEM, ARE NOT TO BE GIVEN TO CUSTOMERS. However, using your own good judgment, you may discuss pertinent information from these pages with a customer. The second part contains information which may be REPRODUCED by the Branch Office at their discretion and given to customers. These pages are not labeled with any restrictive classification. Note that when any abstracts for Type III programs are given to the customer, he must be informed that the disclaimer for Type III programs, as contained in the Newsletter, applies. YOUR COOPERATION IN PROPERLY HANDLING THE CONTENT OF THE IBM INSTALLATION NEWSLETTER IS NECESSARY FOR ITS CONTINUED PUBLICATION.

Changes to the quantity of the Newsletter shipped are to be made by the Administration Manager. (NOTE THAT EXTRA COPIES OF THE NEWSLETTER ARE NOT TO BE ORDERED FOR DISTRIBUTION OF INFORMATION TO CUSTOMERS.) He should update and send the DP Administration Managers' Communication Control card (form number N44-780) to DP Communication Control, DPD HQ. (The card N44-780 replaces N37251.)

Direct inquiries concerning the contents of this Newsletter to:

W. P. Gaffigan, Editor
IBM Installation Newsletter
Systems Engineering Technical Programs
DPD HQ
112 East Post Road
White Plains, New York 10601
Telephone 914 WH9-1900 Extension 4781

- PRL #25 June 23, 1967 -

S20-1510-1	IBM 2915 Airlines Display Terminal Facts Folder	Scrap S20-1510-0
A24-1421-3	IBM 1412 Magnetic Character Reader Model 1	Use A24-1421-2, N24-0316 & N24-0354
A24-3125-2	IBM 1050 Operator's Guide	Use A24-3125-1 & N27-3000
A26-5663-0	IBM 2303 Drum Storage OEMI	NEW
B20-0070-1	Piecework Payrolls for the Garment Industry	Use B20-0070-0
C24-3398-4	IBM S/360 BPS Input/Output 1412/1419 Specifications and Operating Guide	Scrap C24-3398-3, -2, -1, N24-5201, N24-5263, N24-5033, N24-5079, N24-5081, N24-5158, N24-5080 & N24-5202.
C24-5061-1	IBM S/360 BPS, Basic Tape System, System Generation and Maintenance	Scrap C24-5061-0 & N24-5245
C28-6218-1	705 Input/Output Interpretive Program for the IBM 7080 INT 580	Use C28-6218-0 & N28-1044
E20-0285-0	Mechanized Library Procedures for the IBM Advanced Systems Development Division Library Los Gatos, California	NEW
H20-0293-1	IBM System/360 Flowchart (360A-SE-22X) User's Manual	Use H20-0293-0
H20-0301-1	Civil Engineering Coordinate Geometry (COGO) for IBM 1130 Model II (1130-EC-02X) User's Man.	Use H20-0301-0
H20-0352-0	1400 Autocoder to COBOL Conversion Aid Program (360A-SE-19X) Application Description Man.	NEW
J24-1411-3	Utility Programs for IBM 1401 Tape Systems - Specifications	Use J24-1411-2 & N21-0030
K20-0077-0	First National Bank in St. Louis Online Savings and Mortgage Service.	NEW
N20-1032-0	TNL for General Purpose Simulation System/360 Disk Operating System (360A-CS-19X) Operator's Manual Re: H20-0327-0	NEW
N21-5061-0	FORTRAN IV Lang. Specs. Prog. Specs. & O.P. IBM 1401, 1440 & 1460 Re: C24-3322-2 with N21-5051	NEW
N23-0124-0	ENL to IBM Data Comm. Concepts Education Guide Re: R20-4086-0	NEW
N26-0174-0	TNL to A26-5847-2 System/360 Model 20 Functional Characteristics with N26-0176 & N26-0179	NEW
N26-0191-0	IBM 1080 Functional Characteristics Re: A26-3661-1 with N26-0186	NEW
N28-0224	IBM 7090/7094 IBSYS Operating System, Version 13: Input/Output Control System Re: C28-6345-4, -5, with N28-0201 & N28-0176	NEW
N33-9010	TNL to Mod. 20 TPS CTRL & Serv. Re: C24-9000-1	NEW
N28-2244-0	IBM S/360 Basic Programming Support: Sort/Merge Programs Operating Guide Re: C24-3413-2 with N21-5046	NEW
N30-5002	IBM S/360 Operating System: Queued Telecommunications Access Method, Message Control Program Re: C30-2005-0	NEW
N33-9009-0	TNL to Mod. 20 TPS, Utility Programs Re: C26-3808-1	NEW
R20-0050-2	IBM IMPACT Exec. Seminar Education Guide	Use R20-0050-1
R20-9204-1	IBM Data Communication Concepts Course Description	Scrap R20-9204-0
R29-0081-3	FORTRAN IV for IBM System/360 P.I. Chapter 1	Use R29-0081-2
R29-0083-3	FORTRAN IV for IBM System/360 P.I. Chapter 3	Use R29-0083-2
R29-0084-3	FORTRAN IV for IBM System/360 P.I. Chapter 4	Use R29-0084-2
R70-0001-0	9361 Student Guide	NEW
V20-0068	IBM 1620 Programs for Petroleum Exploration (slides)	Abstract only
V20-0069	IBM 1620 Programs for Petroleum Engineering (slides)	Abstract only
X26-5961-4	Physical Planning Template System/360 Model 20	Scrap X26-5961-3
Y70-0001-0	9361 Instructors Guide	NEW
Z77-6364-1	The NOBLIS Computer, its Simulator, a Course Outline and Teaching Notes as used to Introduce System/360 Computer Concepts	Scrap Z77-6364-0
Z77-7146-0	A System/360 Mod. 20 Method of Performing the Large Printing Functions of a 519 End Printer	NEW
Z77-7147-0	Interfacing Tape and Communications IOCS in the System/360 Mod 20	NEW
Z77-7148-0	The Optimization of Condensate in the Production of Gas Wells	NEW

July 14, 1967 Issue No. 67-14

TABLE OF CONTENTS

DOS/360 - ABILITY TO OPEN 7 TRACK TAPE FILES WITH LABELS

Labels on 7 track tapes must be initialized with translate on, data convert off, and even parity, regardless of the mode of the data to be recorded on that file. IOCS reads the labels assuming this condition. IOCS does, however, check the mode setting for density.

DOS/BOS INDEXED SEQUENTIAL FILE MANAGEMENT SYSTEM

The following two notes may affect the use of ISFMS.

1. During an add/retrieve operation, ISFMS does not create new entries in the indices that are maintained. This poses a problem when a file is being extended by adding new records to the end of the file and other records in the file are randomly retrieved. The file is extended, but all new records are added into the data overflow areas only (not by extension of the prime data area). The solution is to define the file twice - once as an add operation, and once as a retrieve operation.

2. A word of caution is necessary when using packed keys. If it becomes necessary to sort the file, the user must reorganize from IS to sequential organization. If this file is later reorganized to IS, the packed keys now contain a sign configuration of "C" not "F" as it was when the file was originally loaded. (The sort created this configuration.) The significance of this is that any searching for these records must contain an identical sign configuration since the search is done on a bit by bit comparison of the key.

DOS/360 RPG - LINE COUNTER SPECIFICATIONS

A significant improvement in throughput speed can be achieved when using the Line Counter specifications in DOS/360 RPG for printers instead of carriage tapes. The Line Counter specifications define the carriage tape for a particular job, and RPG then keeps track internally of where every line is to be printed. This means that RPG never issues a test for carriage overflow on the printer when using the Line Counter specifications.

Because RPG only sets up one I/O area for each file, this means that if you are not using the Line Counter specifications, DOS/360 waits for device end on each print command so that it can then issue the test for carriage overflow. The overlap of other processing and other I/O commands is eliminated while waiting for device end. When using the Line Counter specifications, DOS/360 does not have to wait for device end.

The Line Counter specifications are, in effect, a picture of the carriage tape. Thus, a properly punched carriage tape is required. "Skip to channel..." will be generated as first character forms control.

The above suggestion can only be implemented starting with Version 2 Level 3 of RPG, which was distributed with Release 11 of DOS/360.

PL/I - LOCATING DSA-MAIN IN A PROGRAM WITH OVERLAYS UNDER S/360 DOS

In a program without overlay, the hi-core address, shown on the link edit map, plus up to seven bytes to go to double word boundary, will point to DSA-Main. With overlays, however, it is no longer valid. At the address of the COMREG plus X'24', the address of hi-core for the last phase fetched will be stored. At COMREG plus X'28', the address of hi-core for the entire program will be stored, and this will be the address of the beginning of DSA-Main.

PL/I - VALIDATION OF REGISTER CONTENT UNDER S/360 DOS

Due to the manner in which PL/I manipulates the registers, the registers listed with a program dump on a termination type error may or may not be valid. At the time a program check interrupt occurs, the old PSW and the registers (0-15) are stored in the library workspace labeled on the link edit map IJKZWSI. The contents of register 13 will always point to the last active DSA and thus can be used to chain back through all the active DSA.

S/360-20 IOCS FOR THE 1419

Unless a buffer-full condition or user-issued DSENG occurs in conjunction with the last documents, the 1419 is engaged when the last document is read, causing a dummy read into core.

A solution to this problem is to make a test for the non-existent document, both in the stacker select routine and in the processing routine.

S/360-20 DISK CONTROL PROGRAM POINTERS

1. General registers 8, 24 and 25 are used and not restored by the system loader.

2. The number of tracks allocated to the Core Image Directory or Macro Directory is limited to 10.

S/360-20 RPG POINTERS CORRECTION

Installation Newsletter issue 67-10 contained, on page 22, an article titled "S/360-20 RPG Pointers." Paragraph 2g of this article should read:

> All indicators can be set on or off, including 1P and 0F.

S/360-20 DISK FILE CAPACITY ESTIMATION CORRECTION

Installation Newsletter issue 67-11 contained, on pages 22 through 27, several charts titled "S/360-20 Disk File Capacity Estimation." In the heading for each chart, the statement:

> One/Two Track(s) for Index = Key Length X-X

should be

> One/Two Sector(s) for Index = Key Length X-X

RPG TAX COMPUTATION SUBROUTINE

This subroutine computes federal, state, city and FICA taxes for a weekly payroll (see Exhibit 1). Entry to the routine is "TAX" and input that must be supplied is gross pay "GROS" (7 positions with 2 decimals), number of exemptions "EXMP" (2 positions with no decimals), year to date FICA tax "FICA" (5 positions with 2 decimals), and marital status via indicator 50 (on for married). The four taxes are returned in "FIT," "ST," "CITY," and "FICA."

State taxes are based on the Maryland Tax Structure which parallels the federal logic with the addition of a standard deduction allowance of ten percent of gross pay up to a maximum of $100.00 for a weekly payroll.

City taxes are a fixed percent of gross, and the percentage has been left zero for this routine.

For states other than Maryland and for payroll systems other than weekly, changes in the constants and tables will be necessary to tailor the routine.

The entire routine, including tables, takes 524 bytes and will be less with the exclusion of the city tax routine where it does not apply.

1401 TO S/360 RPG CONVERSION CONSIDERATIONS

The following is a list of items to facilitate conversion of 1400 RPG to S/360 RPG.

1. All fields being edited have to be specified as numeric on the Data Specification Sheet.

2. All fields being zero-suppressed have to be specified as numeric on the Data Specification Sheet.

3. All fields being used in an Arithmetic Operation have to be specified as being numeric on the Data Specification Sheet.

4. Two fields being compared have to be specified as being both alphameric or both numeric on the Data Specification Sheet.

```
*                    RPG TAX COMPUTATION SUBROUTINE                    TAX   10
*                   RPG SUBROUTINE FOR FEDERAL, STATE,                 TAX   20
*                   CITY, AND FICA TAXES (WEEKLY)                      TAX   30
*                   WRITTEN BY R.P.CHAMBERS  042                       TAX   40
*                   JUNE,1967                                          TAX   50
*                                                                      TAX   60
*                      ENTRY - TAX                                     TAX   70
*                                                                      TAX   80
*                      EXTERNAL SYMBOLS                                TAX   90
*                           EXMP  -  NUMBER OF EXEMPTIONS              TAX  100
*                           GROS  -  GROSS PAY                         TAX  110
*                           FIT   -  FEDERAL TAX                       TAX  120
*                           ST    -  STATE TAX (BASED ON MARYLAND)     TAX  130
*                           CITY  -  CITY TAX  (BASED ON 00% OF GROSS) TAX  140
*                           FICA  -  SOCIAL SECURITY TAX               TAX  150
*                                    (YTD FICA SHOULD BE PUT IN FICA   TAX  160
*                                     SO THAT ROUTINE CAN STOP AT      TAX  170
*                                     LIMIT)                           TAX  180
*                           IN50  -  INDICATOR 50 IS ON FOR MARRIED    TAX  190
*                                                                      TAX  200
*                                                                      TAX  210
TAX      START 0                                                       TAX  220
         EXTRN EXMP          LENGTH =  2 BYTES                         TAX  230
         EXTRN GROS          LENGTH =  4 BYTES                         TAX  240
         EXTRN FIT           LENGTH =  3 BYTES                         TAX  250
         EXTRN ST            LENGTH =  3 BYTES                         TAX  260
         EXTRN CITY          LENGTH =  3 BYTES                         TAX  270
         EXTRN FICA          LENGTH =  3 BYTES                         TAX  280
         EXTRN IN50          LENGTH =  1 BYTES                         TAX  290
*                                                                      TAX  300
*                  ROUTINE INITIALIZATION                              TAX  310
*                                                                      TAX  320
         BASR  15,0                                                    TAX  330
         USING *,15                                                    TAX  340
         STH   8,SAV8        SAVE REG.8                                TAX  350
         STH   10,SAVA       SAVE REG.10                               TAX  360
*                                                                      TAX  370
*                  FIRST ROUTINE COMPUTES FEDERAL TAX                  TAX  380
*                                                                      TAX  390
         ZAP   WORK,0(2)          LOAD NUMBER                          TAX  400
         ORG   *-2                OF EXEMPTIONS                        TAX  410
         DC    Y(EXMP)            INTO WORK AND                        TAX  420
         MP    WORK,K1            MULTIPLY BY EXEMP AMT(MINUS)         TAX  430
         AP    WORK,0(4)          ADD GROSS TO                         TAX  440
         ORG   *-2                DETERMINE                            TAX  450
         DC    Y(GROS)            TAXABLE GROSS AND                    TAX  460
         BC    12,NOF             BRANCH TO NOTAX IF ZERO OR MINUS     TAX  470
         LH    8,ADC1             LOAD ADDRESS OF SINGLE TABLE         TAX  480
         CLI   0,X'00'            CHECK FOR MARRIED                    TAX  490
         ORG   *-2                AND LOAD ADDRESS                     TAX  500
         DC    Y(IN50)            OF MARRIED TABLE                     TAX  510
         BC    8,*+8              IF SO, THEN                          TAX  520
12       LH    8,ADC2             GO TO TABLE                          TAX  530
11       BAS   10,LOOK            LOOK UP ROUTINE                      TAX  540
10       ZAP   0(3),WORK+5(3)     PUT TAX IN FIT                       TAX  550
9        ORG   *-4                                                     TAX  560
8        DC    Y(FIT)                                                  TAX  570
7
6
```

Exhibit 1

```
         ORG   *+2                                                  TAX  580
         BC    15,*+10                                              TAX  590
*        SP    FIT,FIT          ZERO FIT (DUMMY INSTR)              TAX  600
NOF      DC    X'FB32'                                              TAX  610
         DC    Y(FIT)                                               TAX  620
         DC    Y(FIT)                                               TAX  630
*                                                                   TAX  640
*              SECOND ROUTINE FINDS STATE TAX                       TAX  650
         ZAP   HOLD,0(2)        LOAD NUMBER OF                      TAX  660
         ORG   *-2              EXEMPTIONS IN HOLD                  TAX  670
         DC    Y(EXMP)                                              TAX  680
         MP    HOLD,K3          MULTIPLY BY -15.50                  TAX  690
         ZAP   WORK,0(4)        LOAD GROSS PAY                      TAX  700
         ORG   *-2              IN WORK                             TAX  710
         DC    Y(GROS)                                              TAX  720
         SP    WORK,K2          SUBTRACT LIMIT                      TAX  730
         BC    2,ST1            IF POSITIVE, DEDUCT LIMIT           TAX  740
         ZAP   WORK,0(4)        OTHERWISE DEDUCT 10%                TAX  750
         ORG   *-2                                                  TAX  760
         DC    Y(GROS)                                              TAX  770
         BC    15,ST2                                               TAX  780
ST1      ZAP   WORK,K2+2(1)     EFFECTIVELY LOAD WORK WITH          TAX  790
         OI    WORK+5,X'10'     LIMIT TO THREE DECIMALS             TAX  800
ST2      AP    WORK,HOLD        ADD EXEMPTION AMOUNT                TAX  810
         AP    WORK,FIVE        ROUND OFF WORK                      TAX  820
         MVO   WORK,WORK(7)     TO TWO DECIMALS                     TAX  830
         OI    WORK+7,X'01'     MAKE EXEMP AMT NEGATIVE             TAX  840
         AP    WORK,0(4)        AND ADD GROSS TO                    TAX  850
         ORG   *-2              DETERMINE THE                       TAX  860
         DC    Y(GROS)          TAXABLE AMOUNT                      TAX  870
         BC    12,NOST          IF ZERO OR MINUS, THEN NO STATE     TAX  880
         LH    8,ADC3           LOAD ADDRESS OF STATE TABLES        TAX  890
         BAS   10,LOOK          AND BRANCH TO LOOK UP ROUTINE       TAX  900
         ZAP   0(3),WORK+5(3)   PUT STATE TAX IN ST                 TAX  910
         ORG   *-4                                                  TAX  920
         DC    Y(ST)                                                TAX  930
         ORG   *+2                                                  TAX  940
         BC    15,*+10                                              TAX  950
*        SP    ST,ST            ZERO STATE TAX (DUMMY INSTR)        TAX  960
NOST     DC    X'FB32'                                              TAX  970
         DC    Y(ST)                                                TAX  980
         DC    Y(ST)                                                TAX  990
*                                                                   TAX 1000
*              THIRD ROUTINE COMPUTES CITY TAX                      TAX 1010
         ZAP   WORK,0(4)        LOAD GROSS PAY                      TAX 1020
         ORG   *-2              IN WORK                             TAX 1030
         DC    Y(GROS)                                              TAX 1040
         MP    WORK,K4          MULTIPLY BY TAX PERCENT             TAX 1050
         MVO   WORK,WORK(7)     ROUND OFF TO                        TAX 1060
         AP    WORK,FIVE        TWO DECIMALS                        TAX 1070
         MVO   WORK,WORK(7)                                         TAX 1080
         ZAP   0(3),WORK+5(3)   PUT CITY TAX IN CITY                TAX 1090
12       ORG   *-4                                                  TAX 1100
11       DC    Y(CITY)                                              TAX 1110
10       ORG   *+2                                                  TAX 1120
*9                                                                  TAX 1130
*8             FOURTH ROUTINE COMPUTES FICA TAX                     TAX 1140
```

Exhibit 1 (continued)

```
        ZAP   WORK,0(4)          LOAD GROSS PAY                    TAX 1150
        ORG   *-2                IN WORK                           TAX 1160
        DC    Y(GROS)                                              TAX 1170
        MP    WORK,K5            MULTIPLY BY FICA PERCENT          TAX 1180
        OI    WORK+5,X'0F'       ROUND OFF TO                      TAX 1190
        AP    WORK(6),FIVE       TWO DECIMALS                      TAX 1200
        MVO   WORK(6),WORK(5)                                      TAX 1210
        AP    0(3),WORK+3(3)     AND ADD TO YTD TAX IN FICA        TAX 1220
        ORG   *-4                                                  TAX 1230
        DC    Y(FICA)                                              TAX 1240
        ORG   *+2                                                  TAX 1250
        SP    0(3),K6            SUBTRACT LIMIT AND                TAX 1260
        ORG   *-4                                                  TAX 1270
        DC    Y(FICA)                                              TAX 1280
        ORG   *+2                                                  TAX 1290
        BC    12,ALL             IF ZERO OR MINUS PUT ALL TAX IN FICA TAX 1300
        SP    WORK+3(3),0(3)     IF PLUS, SUBTRACT EXCESS FROM TAX TAX 1310
        ORG   *-2                                                  TAX 1320
        DC    Y(FICA)                                              TAX 1330
ALL     ZAP   0(3),WORK+3(3)     AND THEN PUT IN FICA              TAX 1340
        ORG   *-4                                                  TAX 1350
        DC    Y(FICA)                                              TAX 1360
        ORG   *+2                                                  TAX 1370
*                                                                  TAX 1380
*            END OF ROUTINES                                       TAX 1390
*                                                                  TAX 1400
        LH    8,SAV8             RESTORE REG.8                     TAX 1410
        LH    10,SAVA            RESTORE REG.10                    TAX 1420
        BCR   15,14              RETURN TO MAIN PROGRAM            TAX 1430
*                                                                  TAX 1440
*                THIS ROUTINE LOOKS UP NEXT LOWEST TABLE           TAX 1450
*            ENTRY(REG.8 POINTS TO ARG. TABLE) TO TAXABLE          TAX 1460
*            GROSS CONTAINED IN WORK. THEN THE TABLE ARG           TAX 1470
*            IS SUBTRACTED TO FIND EXCESS; MULTIPLIED BY           TAX 1480
*            PERCENT AND THE BASE TAX IS THEN ADDED WITH           TAX 1490
*            FINAL TAX RESULT IN WORK.                             TAX 1500
*                                                                  TAX 1510
LOOK    CP    WORK(8),0(3,8)     COMPARE WORK WITH FIRST ARG.      TAX 1520
        BC    2,FIND             IF ARG IS LOWER, GO TO FIND       TAX 1530
        AH    8,*+8              OTHERWISE, INCREMENT TO NEXT ARG. TAX 1540
        BC    15,LOOK            AND LOOK AGAIN                    TAX 1550
        DC    Y(8)                                                 TAX 1560
FIND    SP    WORK,0(3,8)        COMPUTE EXCESS OVER BASE AMT.     TAX 1570
        MP    WORK,6(2,8)        MULTIPLY BY PER CENT              TAX 1580
        MVO   WORK,WORK(7)       ROUND OFF TO TWO DECIMALS         TAX 1590
        MVO   WORK,WORK(7)                                         TAX 1600
        AP    WORK,FIVE                                            TAX 1610
        MVO   WORK,WORK(7)                                         TAX 1620
        AP    WORK,3(3,8)        ADD BASE TAX                      TAX 1630
        BCR   15,10              RETURN TO SUB-PROGRAM ROUTINE     TAX 1640
*                                                                  TAX 1650
*            CONSTANTS AND DATA AREAS                              TAX 1660
*2                                                                 TAX 1670
SAV8    DS    H                                                    TAX 1680
SAVA    DS    H                                                    TAX 1690
K1      DC    X'1350D'           FEDERAL EXEMPTION AMOUNT          TAX 1700
K2      DC    X'10000F'          STATE DEDUCTION LIMIT             TAX 1710
7
6
```

Exhibit 1 (continued)

```
K3          DC    X'15500F'          STATE EXEMPTION AMOUNT          TAX 1720
K4          DC    X'000F'            CITY TAX PERCENTAGE             TAX 1730
K5          DC    X'4400F'           FICA TAX PERCENTAGE             TAX 1740
K6          DC    X'29040F'          FICA TAX LIMIT                  TAX 1750
WORK        DS    CL8                                                TAX 1760
FIVE        DC    X'5F'                                              TAX 1770
HOLD        DS    CL6                                                TAX 1780
*                                                                    TAX 1790
*                  TAX TABLES - WEEKLY                               TAX 1800
*                       1) FEDERAL SINGLE                            TAX 1810
*                                                                    TAX 1820
ADC1        DC    Y(*+2)                                             TAX 1830
            DC    X'21201F'                                          TAX 1840
            DC    X'4085F'                                           TAX 1850
            DC    X'300F'                                            TAX 1860
            DC    X'16901F'                                          TAX 1870
            DC    X'3010F'                                           TAX 1880
            DC    X'250F'                                            TAX 1890
            DC    X'08501F'                                          TAX 1900
            DC    X'1330F'                                           TAX 1910
            DC    X'200F'                                            TAX 1920
            DC    X'02301F'                                          TAX 1930
            DC    X'0276F'                                           TAX 1940
            DC    X'170F'                                            TAX 1950
            DC    X'01301F'                                          TAX 1960
            DC    X'0126F'                                           TAX 1970
            DC    X'150F'                                            TAX 1980
            DC    X'00401F'                                          TAX 1990
            DC    X'00000F'                                          TAX 2000
            DC    X'140F'                                            TAX 2010
            DC    X'00000F'                                          TAX 2020
            DC    X'00000F000F'                                      TAX 2030
*                                                                    TAX 2040
*                       2) FEDERAL MARRIED                           TAX 2050
*                                                                    TAX 2060
ADC2        DC    Y(*+2)                                             TAX 2070
            DC    X'42301F'                                          TAX 2080
            DC    X'8119F'                                           TAX 2090
            DC    X'300F'                                            TAX 2100
            DC    X'34001F'                                          TAX 2110
            DC    X'6044F'                                           TAX 2120
            DC    X'250F'                                            TAX 2130
            DC    X'16901F'                                          TAX 2140
            DC    X'2624F'                                           TAX 2150
            DC    X'200F'                                            TAX 2160
            DC    X'08501F'                                          TAX 2170
            DC    X'1196F'                                           TAX 2180
            DC    X'170F'                                            TAX 2190
            DC    X'02301F'                                          TAX 2200
            DC    X'0266F'                                           TAX 2210
            DC    X'150F'                                            TAX 2220
            DC    X'00401F'                                          TAX 2230
2           DC    X'00000F'                                          TAX 2240
1           DC    X'140F'                                            TAX 2250
0           DC    X'00000F'                                          TAX 2260
9           DC    X'00000F000F'                                      TAX 2270
8                                                                    TAX 2280
7
5
```

Exhibit 1 (continued)

```
*                          3) STATE                              TAX 2290
*                                                                TAX 2300
ADC3    DC      Y(*+2)                                           TAX 2310
        DC      X'05701F'                                        TAX 2320
        DC      X'0208F'                                         TAX 2330
        DC      X'60F'                                           TAX 2340
        DC      X'03801F'                                        TAX 2350
        DC      X'0115F'                                         TAX 2360
        DC      X'48F'                                           TAX 2370
        DC      X'01901F'                                        TAX 2380
        DC      X'0046F'                                         TAX 2390
        DC      X'36F'                                           TAX 2400
        DC      X'00000F'                                        TAX 2410
        DC      X'00000F'                                        TAX 2420
        DC      X'000F'                                          TAX 2430
        END     TAX                                              TAX 2440
```

Exhibit 1 (continued)

5. Field name on calculation specifications should be specified on all add, subtract, multiply and divide operations. (No "so-called chaining" can be handled by the translator.) Figure 31 of the 1401 RPG Manual shows how 1401 will accept a blank field name. This type of operation will cause a problem when converting to S/360 RPG.

   This is also true on the Data Specification Sheets. Field name should always be specified. Figure 18 of the 1401 RPG Manual is an example of this operation. Field 0 should be repeated on the line below.

6. All edit words should have a zero-suppress character specified; otherwise, full zero-suppress will result.

7. In 1400 RPG, the first Data Record causes a control break. In S/360 RPG, "Run-In" control breaks are automatically suppressed. Therefore, it would be a good idea to avoid printing headings or performing calculations on a "Run-In" condition since it will not be available to you on S/360 RPG.

8. All 1P lines should have a skip to channel 1 before printing, otherwise first page headings will print on the same page as the linkage editor when performing S/360 compile and go operations.

9. Word entries being used as a constant must be specified with an F in column 1 (format type). A K was accepted by the 1400 assembler.

10. Because overflow is implemented in a different way in the S/360 RPG, output lines conditioned by both OF and other conditions in an OR relationship must be programmed as being mutually exclusive.

    If, in the S/360, the overflow and L1 (level one indicator) occur at the same time, the line prints twice, thereby, possibly producing an erroneous report.

    To prevent this situation, the line would have to be coded in a S/360 program as OFNL1 for the Field Output Condition. This would have to be added by the programmer after conversion.

11. A numeric field which has a value of +0 or a -0 turns on only the zero indicator; the plus and minus indicators are not turned on. There is no -0 value in the S/360; zero is always +0.

12. In S/360, blank numeric fields are read into the system as zeros. Therefore, it is not possible to distinguish between a blank or zero input field for the purpose of setting on a zero field indicator.

    Alphameric fields containing blanks are read in as blanks and can be used to set on a blank field indicator. Zeros in alphameric fields are read in as character zeros and cannot be used to set on a zero field indicator.

    Alphameric fields cannot be tested for plus or minus conditions.

13. The Translator program will not translate sense switch functions. Field indicators should be used to control the operations that would normally be controlled by sense switches.

14. A WORDXX that defines a numeric literal greater than 15 characters will not be translated because numeric fields over 15 characters are not allowed on S/360 RPG.

15. 1400 RPG Format Specifications allow a maximum of 28 character literals or edit words. S/360 Output Specifications permit a maximum of only 26 character literals or edit words.

16. The Decimal Control feature included in the 1400 Expanded Print Edit Feature is not included in S/360 editing. In S/360 RPG, the decimal point is removed during a zero suppress operation if it is to the right of the leftmost significant digit and to the left of the zero suppression character.

17. Any constant information in the leftmost positions of the edit word is regarded as part of the body portion and is suppressed if it is to the left of the leftmost significant digits. CR or minus signs should always be in the rightmost position of the edit word.

18. PAGENO should be defined as a field length of 004, because in S/360 it must be a full word (4 bytes).

S/360-44 PROGRAMMING SYSTEM

Before referencing a member just added to a M44PS directoried data set, one must insure the latter has been closed.

After adding any routine to SDSREL or any member to a directoried D/S and prior to referencing it, end the job with a /& and place an operator ACCESS command in front of the next JOB card. Note that the operator command has no // and the name begins in column 1.

```
SYSREL ACCESS SDSREL,SAME=SDSABS
```

S/360-44 PROGRAMMING SYSTEM SAMPLE PROGRAM

The released sample program for the Model 44 programming system does not utilize carriage control. Insert the following card after the // SYSOPT card. // LABEL, CTLASA ASA CHAR = etc.

S/360-44 FORTRAN SUBROUTINE - LINK

There is an error in the M44 FORTRAN run time subroutine named LINK. A fix is available at your regional Field Systems Center.

A phase cannot be fetched, and this results in the supervisor message GA 081 xxxxxxxxPHASE CANNOT BE FETCHED where xxxxxxxx are garbage characters.

S/360-44 UTILITY

There is an error in the M44 Utility CPYMEM* which affects adding members to a Directoried Data Set such as SDSREL.

If the total number of names in the output directory including aliases is a multiple of the blocking factor, no more members can be added. Temporarily, one can punch out the last member, delete this member from the directoried D/S, CONDENSE the directoried D/S, add the just deleted member along with the new members.

TYPE III PROGRAMS FOR S/360 USERS

Often as you review Type III abstracts, there is a tendency to bypass 1401/1620 programs if you are involved only with S/360. If the S/360 has emulation, however, you may be missing programs which are applicable to your account.

An example of this type of program is the Create Print Images on Tape. This abstract appeared in issue 67-09 of the Installation Newsletter. This particular program provides the capability for creating a print image tape during a processing run rather than going to printer, thus pro-

viding faster throughput for printer bound programs. This tape can then be used either under multi-programming or on a system using dual printers to overlap printing during subsequent jobs. The saving to the S/360 customer in total processing time can be substantial. This is just one example of a 1401 Type III program which is applicable to a S/360 customer.

NEW TYPE III PROGRAMS *

The following are the abstracts of programs which have been recently made available from the Type III library.

The program, along with its complete abstract, will be incorporated in subsequent issues of the Catalog of Programs.

Programs may be obtained by submitting a properly completed "General Program Request Card" (Form Number 120-1145-1) to the Program Information Department, 40 Saw Mill River Road, Hawthorne, New York, 10532.

These programs and their related documentation are distributed by IBM in the author's original form and have not been subjected to any formal testing.

Any discussion of Type III Programs must emphasize the following points:

1. Type III programs are not part of the IBM product line, as are Programming Systems (Type I) or Application Programs (Type II).
2. Type III programs have not been subjected to any formal product test.
3. Recipients of Type III programs are expected to make the final evaluation as to the usefulness of the programs in their own environment.
4. There is no committed maintenance for Type III programs. However, any changes the author chooses to make will be announced in subsequent issues of the Catalog of Programs.

* NOTE: THE CUSTOMER MUST BE INFORMED THAT THE ABOVE APPLIES TO ANY OF THE FOLLOWING TYPE III ABSTRACTS FURNISHED TO HIM.

S/360

HOUSTON AUTOMATIC SPOOLING PRIORITY SYSTEM. This system is an automatic SYSIN/SYSOUT/SYSPUNCH SPOOLing package with the ability to operate an essentially unlimited number of peripheral devices concurrently in conjunction with OS/360 processing utilizing a variable number of 2311 and/or 2314 direct-access modules for intermediate storage.

HASP operates with no modifications to OS/360. Any option 2 installation may use HASP by simply adding it to his program library and adding one user SVC routine to OS/360. HASP may then be invoked (Optionally) at any time by standard partition loading procedures.

HASP has been designed to operate as independently as possible of OS/360 release levels and, excluding major design changes, should adapt to new releases of the Operating System with no modifications.

Through the use of HASPGEN, the package may be adapted to virtually any configuration and/or user requirement.

System Configuration: The program source is in OS/360 assembly language, and minimum core requirements are less than 27K.

Ordering Procedure: Order File Number 360D-00.1.008. The program material can be obtained on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR). The requester must indicate whether a 9 track or 7 track DTR is required. If not specified, a 9 track DTR will be provided. No tape submittal is required - the DTR will be provided by the Library.

MODIFICATION OF S/360 WHOLESALE IMPACT LIBRARY FOR USE WITH A 16K - 4 TAPE S/360. The S/360 Wholesale Impact computer library has been designed to provide a user with information essential for his inventory management. The information enables him to establish an inventory strategy which minimizes costs at any preselected service level. The Wholesale Impact Library (360A-DW-05X) now exists as a Type II application program and consists of eight programs. Five of these programs are run infrequently, usually during initial use of the programs and again once every six months to a year. The remaining programs, however, are used much more frequently, perhaps several times each week.

Current machine requirements for use of these programs are a minimum of 16K storage and one 2311 disk drive. Thus, the tape oriented user who is interested in inventory management is unable to benefit from this series of application programs.

The purpose of this report is to describe the modifications which have been made to two of the programs in the wholesale IMPACT library (Type II program - 360A-DW-05X) which allow them to operate under a tape oriented system. The programs which have been modified are SVA (Service Point and Variable Interval Allocation) and AP (Allocation Print). These are two of the most frequently used programs in the library. Modifications have been made to allow a user with a minimum of 16K storage and four tape units to use these two programs (1). Initialization of inventory data will still have to be done on a disk oriented system, but once this has been accomplished the user will be able to do his routine processing on his own tape oriented system. With these modified programs, the tape oriented user can now benefit from these application programs.

Ordering Procedure: Order File Number 360D-25.2.002. To obtain program material, submit one 2400 foot reel of magnetic tape. Specify whether 7 or 9 track recording is required. If not specified, 9 track recording will be used. The required tape may be ordered from IBM or supplied with your request for the program.

IBM 360/20 RPG/BAL RELOCATABLE FEDERAL WITHHOLDING TAX ROUTINES. The calculation of Federal withholding tax is standard with all installations doing payroll. Many installations have found it difficult to code this routine in RPG using a reasonable amount of core which only compounds the core limitations on payroll programs, which, by their nature, are already complex. Routines for weekly, bi-weekly, semi-monthly and monthly payrolls are included which will reduce considerably the duplication of effort by systems engineers and programmers of coding and debugging this problem.

The routines were written in 360/20 BAL and will calculate withholding tax to $999.99. The programmer provides gross, number of exemptions and marital status to the routines and uses the RPG EXIT operation. Machine configuration: 360/20 no special features; Programming System Requirement: 360/20 RPG; Source Language: 360/20 BAL; Storage Requirement: 387 bytes per routine.

Ordering Procedure: Order File Number 360D-19-3.001. The program material can be obtained in card form or on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR). The requester must indicate whether a 9 track or 7 track DTR is required. If not specified, a 9 track DTR will be provided. No tape submittal is required - the DTR will be provided by the Library.

1130

IBM 1130 PHOTOCOMPOSITION PROGRAM. This program provides the user of photocomposition equipment a means of generating fully justified copy in a variety of formats for a number of different photounits. The program has been designed to work in conjunction with any of the "Output Modules" currently available or other modules which may be written to meet a need not currently satisfied by existing modules. The documentation supplied provides sufficient information in the way of program format to permit the writing of new modules.

When ordering the IBM 1130 Photocomposition Program the user must order, in addition, the following programs: (1) IBM 1130 Type Composition Program (1130-DP-04X) and (2) an ap-

propriate Output Module (see program names and file number below). For those users writing a new module, this can serve as an example.

Related Programs:

Title	File Number
1130 Photocomposition Grid/ Face Utility Program	1130-00.4.002
Linofilm Output Module for IBM 1130 Photocomposition Program	1130-06.5.002
Photon 513 Output Module for IBM 1130 Photocomposition Program	1130-06.5.003
Photon 713 Output Module for IBM 1130 Photocomposition Program	1130-06.5.004

System Configuration: As a minimum an 1131 Processor Model 2B with 2315 Disk Cartridge, 1442 Card Read Punch Model 6 or 7, and Basic RPQ's: RPQ 834398 - Basic Interface (required to attach any number of PTR's and PTP's); RPQ 834399 - Paper Tape Attachment (required to attach any number of PTR's and PTP's). Additional information regarding machine configuration and RPQ requirements is contained in Programming Announcement Letter P66-107 relating to the 1130 Type Composition Program (1130-DP-04X). Source Language - 1130 Assembler Language.

Ordering Procedure: Order File Number 1130-06.5.001. Distribution will be in card form only.

1130 PHOTOCOMPOSITION GRID/FACE UTILITY. This program is designed as an installation aid to Linofilm and Photon 513 users of 1130 Photocomposition. Photon 713 users are referred to the Photon 713 Output Module (see file number below) for the procedure to be used in storing face tables. The Utility Program stores on the disk cartridge all character width and point size data required by the 1130 Photocomposition Processor to create a justified line. Input to the 1130 Photocomposition Utility may be entered in cards, 8-channel paper tape, or via the console typewriter.

Included with the basic information are instructions and worksheets to facilitate the gathering and entering of required data and the necessary instructions for operating the program. Program load procedures and special support programs (typesetting Monitor and Skeleton) are available as basic documentation supplied with the 1130 Photocomposition Program and must be ordered at this time (see file number below).

Related Programs:

Title	File Number
IBM 1130 Photocomposition Program	1130-06.5.001
Linofilm Output Module for IBM 1130 Photocomposition Program	1130-06.5.002
Photon 513 Output Module for IBM 1130 Photocomposition Program	1130-06.5.003
Photon 713 Output Module for IBM 1130 Photocomposition Program	1130-06.5.004

System Configuration: 1131 Model 2B and either 1134 paper tape reader or 1442 Model 6 or 7 card read/punch.

Ordering Procedure: Order File Number 1130-00.4.002. Distribution will be in card form only.

LINOFILM OUTPUT MODULE FOR IBM 1130 PHOTOCOMPOSITION PROGRAM. The Linofilm Output Module has been designed to work in conjunction with the IBM 1130 Photocomposition Program to produce a three frame 6-channel paper tape punched in a format capable of driving the Linofilm photounit equipped with the 6-channel paper tape adapter. Either the 18 Grid Linofilm Unit or the 28 Grid Linofilm Unit may be driven by the tapes generated by this Output Module. The Linofilm Output Module, in conjunction with the IBM 1130 Photocomposition Program, permits the user to process unjustified tape, prepared on a standard 6-channel tape perforator, and thereby derive an output tape which contains lines of justified copy, with appropriate photounit function codes, as determined by commands entered with the unjustified input.

Users must order both the IBM 1130 Photocomposition Program (1130-06.5.001) and the 1130 Type Composition (1130-DP-04X) when ordering this Output Module.

System Configuration: As a minimum an 1131 Processor Model 2B with 2315 Disk Cartridge, 1442 Card Read Punch Model 6 or 7, and Basic RPQ's. RPQ 834398: Basic Interface (required to attach any number of PTR's and PTP's); RPQ 834399: Paper Tape Attachment (required to attach any number of PTR's and PTP's). Additional information regarding machine configuration and RPQ requirements can be found in the Catalog of Programs for the IBM 1130 and 1800 (20-1630) under the description of the 1130 Type Composition Program (1130-DP-04X). Source Language: 1130 Assembler Language.

Ordering Procedure: Order File Number 1130-06.5.002. Distribution will be in card form only.

PHOTON 513 OUTPUT MODULE FOR IBM 1130 PHOTOCOMPOSITION PROGRAM. The Photon 513 Output Module has been designed to work in conjunction with the IBM 1130 Photocomposition Program, to produce 6-channel paper tape, punched in a format capable of driving a Photon 513 photocomposing machine. Since the Photon 513 is available as either a forward or backward reading unit, the material supplied with this program is designed to permit the user to implement a system which will drive either or both. Users must order both the 1130 Photocomposition Program (1130-06.5.001) and the 1130 Type Composition Program (1130-DP-04X) when ordering this Output Module.

System Configuration: As a minimum an 1131 Processor Model 2B with 2315 Disk Cartridge, 1442 Card Read Punch Model 6 or 7, and Basic RPQ's: RPQ 834398 - Basic Interface (required to attach any number of PTR's and PTP's); RPQ 834399 - Paper Tape Attachment (required to attach any number of PTR's and PTP's). Additional information regarding machine configuration and RPQ requirements can be found in the Catalog of Programs for the IBM 1130 and 1800 (20-1630) under the description of the 1130 Type Composition Program (1130-DP-04X). Source Language: 1130 Assembler Language.

Ordering Procedure: Order File Number 1130-06.5.003. Distribution will be in card form only.

PHOTON 713 OUTPUT MODULE FOR 1130 PHOTOCOMPOSITION. The Photon 713 Output Module has been designed to work in conjunction with the 1130 Photocomposition Program to produce 6-channel paper tape, punched in a format capable of driving a Texmaster (R) photocomposing machine. The 6-channel format produced meets the input requirements of photo units with production numbers greater than 50. References in the documentation provide the information necessary to modify the program for production numbers lower than 50. Users must order both the 1130 Photocomposition Program (1130-06.5.001) and the 1130 Type Composition Program (1130-DP-04X) when ordering this Output Module.

System Configuration: As a minimum an 1131 Processor Model 2B with 2315 Disk Cartridge, 1442 Card Read Punch Model 6 or 7, and Basic RPQ's. RPQ 834398 - Basic Interface (required to attach any number of PTR's and PTP's); RPQ 834399 - Paper Tape Attachment (required to attach any number of PTR's and PTP's). Additional information regarding machine configuration and RPQ requirements can be found in the Catalog of Programs for the IBM 1130 and 1800 (20-1630) under the description of the 1130 Type Composition Program (1130-DP-04X). Source Language: 1130 Assembler Language.

Ordering Procedure: Order File Number 1130-06.5.004. Distribution will be in card form only.

1800

1800 TSX CONTROL CARD GENERATOR. This program generates all control cards necessary to commence a system generation for the 1800 TSX Operating System (1800-OS-001). All input data is defined on pre-printed keypunch forms designed to minimize and centralize all required parameters. The control card output consists of: (1) Non-Process Monitor DUP control cards, (2) System Loader Assignment and DEDIT cards, (3) TASK group 1 and 2 equate EQU cards, (4) System Director equate EQU cards. Major advantages of the program

are the guaranteeing of proper formatting and punching of control cards and the provision for consistent information in the many inter-dependent parameters required by the TSX system.

System Configuration: The source language is 1800 Assembler. The object program executes under the control of System Generation TASK and requires the following machine configuration: 1801 or 1802 PC with 16K or 32K core storage, 2310 A1 Disk, 1442 Card Read-Punch and one 1053 Typewriter.

Ordering Procedure: Order File Number 1800-03.0.001. Distribution will be in card form only.

IBM Installation Newsletter

January 26, 1968

Issue No. 68-01

CONTENTS

(G means Green Section)

*Requires Immediate Attention

For IBM Internal Use Only

DPD HQ, White Plains, N. Y.

SELECTED SHORT SUBJECTS

The purpose of Selected Short Subjects is to bring together and highlight concise, factual and timely information which will indicate that action is to be taken by the IBM representative whose accounts are affected.

1. OS/360 Release 14 Planning Information-Clarification

Installation Newsletter issue 67-25 contained an article with the above title. The second sentence under item 7, on page 2, should read "Console messages have been rewritten to improve clarity and unit affinity is supported".

2. Use of the Product and Support Requirement Request Form, 120-1702-0

When submitting the Product and Support Requirement Request, the branch office should retain their copy and forward the entire remaining copies to their region for evaluation and comment. The region will respond to the branch office on the response copy and forward copy 3 to the Director of Systems Marketing, DPD HQ. It is important that this procedure be followed so that the DP Dvision will be aware of the response and evaluation made on each Product and Support Requirement Request.

3. S/360 New QNC Character Set-1416 Chain Cartridge

Installation Newsletter issue 67-26 contained an article on the above subject on page 11. It included a statement "See the Type Catalog in the Sales Manual for further details".

A memo has been received from the field indicating that there is only one comma on the train. Customers have the option to substitute type slugs.

See the paragraph "Type Slug Substitutions" on page 72 of the Type Catalog in the Sales Manual.

4. Teleprocessing Custom Terminals

If you have a unique terminal application requirement, you should be aware of a slide presentation, with narrative, on custom terminals.

It has been distributed to Branch Offices, Education Centers and Regional Special Equipment Departments. A limited supply of additional sets is available (there is a charge) from Audio Visual Services, DPD HQ. The form number is Z20-5002. The item is not available from Mechanicsburg. For further information see the abstract in the MAPS binder, page F 20-02-01. 02.

5. Glossary of Communications Common Carrier Terminals

If you are involved in the communications area, this glossary (520-1563) may be useful to you in your contacts with Common Carrier representatives and customers and propects who speak the language of communications.

6. Classification Code Index Added to Catalog of Programs

A Classification Code Index for Contributed Programs has been added to the December 1967 edition of the Catalog of Programs for IBM System/360 (C20-1619-4) and Catalog of Programs for IBM 1130 Computer System and IBM 1800 Data Acquisition and Control System (C20-1630-3).

The new Classification Code Index will allow a user to locate programs and subroutines within programs by Classification Code. The index is composed of the Primary Classification Code and Secondary Classification Codes appearing within the body of the Contributed Program abstract.

The catalogs will be available from Mechanicsburg Distribution Center when announced in the PRL (Publications Release Letter).

OS/360 RELEASE 14 EMPHASIS ON UPGRADING

Release 14 of OS/360, as announced in Program Announcement P68-7, is now available. A major emphasis is being placed on upgrading all OS/360 installations to this release level. An analysis as of December 1967 shows over 50% of OS/360 users to be using OS Release 11

or earlier. For many reasons, it is important to both users and IBM to upgrade to Release 14 as soon as possible:

- Take advantage of APAR corrections. Over 900 APAR's were fixed in Releases 12, 13 and 14.

- MFT Version I Dependency. As stated in P68-7, the next OS release will have MFT-II replacing MFT-I. Users who depend on the MFT-I internal logic (HASP users, for example) may find greater difficulty in upgrading to future releases. Establish a firm base if future upgrading may be a problem.

- Take advantage of improved functions. Releases 13 and 14 included many operational improvements, plus many other developments. (See P67-114 and P68-7). FORTRAN H, Version II gives about 30% faster compilation.

This emphasis on upgrading to Release makes the prompt identification and correction of Release 14 problems particularly important. Please report problems problems promptly through the normal FE channels. A special effort is planned to provide PTF response to urgent problems found on this release.

OS/360 RELEASE 14 MVT PROCLIB CATALOGING

The following is provided for planning purposes. Actual availability of OS/360 Releases are as announced through the official IBM announcement procedures.

The following changes are being made in Release 14 for MVT.

PROCLIB is automatically cataloged by the system at IPL time. If PROCLIB was cataloged prior to IPL time and the user specifies a new UNIT via the SET DATE Command, then PROCLIB is uncataloged and then recataloged to reflect the unit specified by the SET DATE command.

OS/360 MVT JOB STEP TIMING ERROR CORRECTION

The following is a field contribution.

There is a logic error in the CSECT IEAQNU00 in the nucleus (OS/360 Release 13).

The effect this logic error has on Job/Step timing ranges from none to extreme variation. It is totally dependent on the environment within the system, the TCB ready queue and the time queue at any given time.

The following correction will give more consistent results within the framework of the current MVT concurrent multiple task priority timing mechanism, that is, a Job Step executed in the same environment will have some repeatability of its execution time. However, the same Job Step executed in a different environment will not produce the same exeuction time.

The correction involves changing the 584th statement in the macro IEAQNU of SYS1.GENLIB (Release 13) from

```
        BCR     8,XSBR      . RETURN IF NOT.
to
        BZ      DJS02       . BR IF NOT -
                              DON'T ENQ/DQ.
```

This correction should be made before Stage II of System Generation, but can be made after the system has been generated by reassembling the macro and relinkediting the nucleus.

In essence, the correction prevents the Job Step Timing Subroutine from abandoning its ENQ/DEQ loop when it encounters a TCB that has no TQE.

The correction can be inserted into SYS1.GENLIB by punching the member IEAQNU with IEBPTPCH, correcting the 584th card, and replacing the corrected macro in the GENLIB with IEBUPDTE.

It has been interpreted from the system specification that system overhead would be charged to the highest priority non-system task. This is not the case. System overhead is not accounted for presently. Hence, when running only one initiator, task time is less than real time.

OS/360 QTAM - CHOOSING POLLING INTERVAL FOR RECEIVE OVER SEND PRIORITY

The following contribution is intended to provide some assistance and guidance in choosing a polling interval when using OS/360 QTAM. The following information resulted during an effort to develop an analytical method for determining message turnaround time in a partitioned system. There has been no official IBM test and potential users are expected to evaluate its usefulness before implementing.

When choosing a polling interval for receive over send priority, an interval equal to the average time to send out a message is not a good choice. During this interval, when no messages are to be sent, the system is prevented from servicing the arriving messages. Thus, the next message to be sent must be delayed by the time to service the arriving messages. In general, when choosing an interval, it is best to choose an interval E where E is the smallest positive number which will effectively reduce constant polling. E should be considerably less than the average time to send out a message. (See the formula shown in Exhibit 1).

The wait Time for a message being sent out is equal to:

$$\frac{\lambda_1(\alpha_1^2 + \mathrm{Var}\alpha_1) + \lambda_2(\alpha_2^2 + \mathrm{Var}\alpha_2)}{2(1 - \lambda_1\alpha_1)(1 - \lambda_1\alpha_1 - \lambda_2\alpha_2)} + \lambda_1 E\alpha_1$$

where λ_1 is the arrival rate of messages

α_1 is the service time for the arriving messages

λ_2 is the departure rate of messages

α_2 is the service time for the outgoing messages

$\mathrm{Var}\alpha_1$ is the variance of the service time α_1

$\mathrm{Var}\alpha_2$ is the variance of the service

Another instance is the case where an interval, I is chosen equal to k times the service time of an outgoing message where k is greater than 1. In this case, the better choice of an interval would be (k - 1) +E, following the above reasoning. However, a large value of k would cause a significant delay in servicing the arriving messages and, thus, force larger queues at the terminals. As the value of k becomes larger, this system approaches send over receive priority.

An illustration of the above system is as follows. Messages are arriving on the line at the rate of .0015 msgs/ms. Messages are departing at the rate of .0015 msgs/ms. Messages are departing at the rate of .0015 msgs/ms. The positive and negative polling times are 18.643 ms, the line speed is 600 characters/sec. Simulation shows that with a polling interval of 200 ms, the waiting time for outgoing messages (before they can be sent) is 239 ms. With a small polling interval, the waiting time for outgoing messages is 212 ms. Note that a 200 ms polling interval also delays the servicing of incoming messages. It should also be noted that the choice of send = receive priority decreases the waiting time to 130 ms. With a choice of send over recieve priority, the waiting time decreases to 60.6 ms.

DOS/360 DATA FILE CONVERSION TO OS/360

Installation Newsletter 67-21, dated October 20, 1967, contained an article with the above title, on page 3. This article contains additional information on the subject.

In DOS files the file name may be 17 characters, including embedded and trailing blanks. In OS, the label data-set name must have no embedded blanks and must not have more than 8 characters between periods.

e.g.

File Name	DOS	OS
MASTER FILE	OK	NO
MASTER.FILE	OK	OK
MASTERFILE	OK	NO

DOS/360 RELEASE 13-LOAD AND FETCH OVERHEAD REDUCTION TECHNIQUE

The following technique is a field contribution. It has not been submitted to any formal IBM test and its usefulness should be evaluated in the intended environment. No IBM maintenance will be available. The technique has not been tested with DOS/360 Release 14. See the author's NOTE below.

The following is a description of a patch made to the DOS/360 supervisor (Release 13) to eliminate much of the system overhead involved with the LOAD and FETCH functions of DOS/360. This has been submitted as a Product Support Requirement Request for DOS/360.

When a background program is started, Job Control builds a phase directory of up to 144 phases which begin with the same first 4 characters as your main phase name. The directory records contain 18 phases each, so if your program contains 18 phases or less, they will be on the first directory record.

The normal LOAD/FETCH operation is to seek to the proper directory (phase directory for BG, FGP directory for FG, etc), read a record and search it for the phase name. The read is repeated until the phase name is found and then a seek is given to the proper core library location so it can be loaded. In a program with many overlays, the probability is high that the proper directory record is already in core from the previous load or fetch. The patch changes the fetch operation to check the directory read area (Physical Transient area) to see if the first four characters match your load/fetch name. If they match, the record in core is searched before going to read the directory. The patch requires 24 additional bytes in the DOS/360 supervisor.

This modification, see Exhibit 1, has reduced run time in one heavily overlayed program from 1/2 hour to 15 minutes, a savings of 50% for this job. It must be noted that jobs which have no overlays, or very few, will not be speeded up appreciably, if at all.

NOTE:

This patch is operational on DOS/360 Release 13 and supervisor with MPS = YES, and has not been subjected to extensive testing under all conditions. Users should implement it with care. The patch will most likely be highly release sensitive, and may not even be feasible with future releases of DOS/360.

DOS/360 TELEPROCESSING CONSIDERATIONS

There are three situations in a DOS/360 system which might temporarily prevent the high priority teleprocessing gaining control of the CPU. These are:

1) During message typing.
2) During execution of error recovery procedures.
3) During program of overlay fetch.

The following further defines these situations:

1. When DOS types an ERP (Error Recovery Procedure) message which does not require an operator response, it goes into an instruction loop testing the typewriter CCB until the typing is completed. During this time the Teleprocessing program cannot get control of the CPU because the transient routine for the typewriter, which operates as part of the higher priority supervisor, retains control. When DOS types an ERP message requiring operator response, it goes into WAIT state with I/O interrupts masked until that response is received. If an operator is not in attendance, the Teleprocessing program will wait to regain control of the CPU from the transient routine until the operator responds.

2. When DOS error recovery transient routines execute I/O, they go into an instruction loop testing the CCB for the device executing the I/O. During this time the Teleprocessing program cannot gain control of the CPU because the transient routine, which operates as part of the higher priority supervisor, does not relinquish control. If individual I/O operations are long, or if a number of I/O operations are required in order to satisfy the ERP requirements (such as with the 2321), the Teleprocessing program may be locked out for some time.

3. When DOS fetches programs or overlays from disk, the disk read commands are followed by an instruction loop which tests the CCB for the DASD device. Thus, the fetch routine does not relinquish control to the Teleprocessing program while executing I/O. Fetching of a large program or overlay may lock out the Teleprocessing for a considerable length of time.

DOS/360 RELEASE 13-LOAD AND FETCH OVERHEAD REDUCTION TECHNIQUE

MODIFICATION TO DOS RELEASE 13 SGDFCH MACRO

```
         MVI    RECNO+1, NODIR                                    H4530058
*   ADDITION1 12-10-67    CHECK FIRST FOUR CHAR. OF PTA FOR EQUAL
PATCH01  LR     RE,R9      LOAD DSECT REG.
         CLC    0(4,R9),DIRNAME    CHECK FIRST 4 FOR EQUAL
         BNE    RDDIR2     GO READ DIRECTORY AS NORMAL
         MVI    PATCH03+1,X'80'    EQUAL - SET NOP TO BE RDDIR2
         B      RDDIR2+8           IN CASE END DIRECTORY FST TIME
*   END FIRST   ADDITION
RDDIR2   BAL    RC,READ UPDT                                      H4530059
*   ADDITION 2 12-10-67
PATCH02  MCI    PATCH03+1,X'00'    SET TO NOP AFTER FIRST TIME
*   END  ADDITION2
         LR     RE,R9                                             H4530060
         LA     RC,DIRLGH(0,R9)                                   H4530061
LOOKUP   CLI    DIRNAME,ENDSGN                                    H4530062
*   ADDITION3 12-10-67
PATCH03  NOP    RDDIR2     IF END DIRECTORY FST. TIME
*   END ADDITION 3
         AIF    (NOT &BG20).LGD01                                 H4530063
```

The release 14 SGDFCH macro is similar, with different sequence numbering. This patch has not been tested on release 14 as yet.

Exhibit 1

When a Teleprocessing program is unable to gain control of the CPU for a number of seconds for one of the above reasons, the Teleprocessing program will be affected in either, or both, of two ways: First, response time will temporarily increase radically. Terminal operators at 2260, 2740 or other terminals may wait 5, 10, 15 or more seconds for the Teleprocessing program to regain control and respond to their input messages, depending on the nature of the Teleprocessing program lockout. Second, programs such as QTAM, which rely on the PCI interrupt, become susceptible to losing incoming data. During message receiving, PCI interrupts indicate the need for additonal core buffer areas to receive the remainder of the message. With the exception of a system waiting for operator response, the above supervisor lockouts do not prevent recognition of the PCI interrupt. However, the routines which must be executed in order to supply an additional buffer, or buffers, are located within the QTAM partition. If that partition cannot regain control in time, input data will be lost. QTAM will recognize this fact and will provide an "insufficient buffers" message on the console. The QTAM program will then have to request a retransmission of the message. Depending on the terminal configuration, the operator may have to reload cards, reposition paper tape, or rekey the message. In addition to causing operator inconvenience this retransmission will, in the case of long messages, require a significant amount of line time.

Although there is no single solution, a number of partial solutions are discussed below:

1) BTAM users should not use the console typewriter for communication with the operator. Instead, a local terminal should be used for this purpose.

2) A user modification to BTAM or QTAM to eliminate or reduce the ERP messages would reduce the conflict for the console typewriter and, therefore, reduce the number of BTAM or QTAM lockouts. This would result in some reduction in line terminal error information. Increasing all threshold values would also reduce console typewriter conflicts by reducing the number of "threshold exceeded" messages.

3) A much better way to reduce QTAM use of the typewriter is to make use of the operator control terminal module. Use of this module causes QTAM ERP messages to print on a selected 1050 or 2740 terminal. Experience to date indicates that this is highly desirable with DOS QTAM.

4) While the Teleprocessing program is running, the installation should arrange to have an operator in attendance at all times.

5) Reducing the number of devices in use while the Teleprocessing program is active would minimize conflict for the ERP transient area. This may, in some cases, result in scheduling problems, or reduction of system performance.

6) Reducing the number of partitions in use while the Teleprocessing program is active would minimize conflict for the typewriter and would also reduce the effect of the fetch routine. This, may be difficult or impossible to do without losing one of the major benefits of DOS BTAM or QTAM- the ability to multiprogram these Teleprocessing programs with other jobs.

7) Increasing the size of QTAM buffers will reduce the number of retransmissions by lengthening the time before a response to the PCI interrupt is required. This is a small sacrifice in small QTAM systems but will have a significant effect on core storage requirements for larger QTAM systems. With certain types of terminals, the use of a large enough buffer will completely eliminate the possibility of losing data due to failure to respond in time to PCI interrupts. For example, keyed input data from a 1050 which is using the auto EOB feature cannot exceed a 130 character block. Similarly, input data from 2260 terminals attached to a 2848 Model 1 cannot exceed a 240 character block.

8) Increasing the size of the BUFNO parameter in the DTFQT will also reduce the number of retransmissions. If more buffers are preallocated to each line, then failure to respond quickly to a PCI interrupt is not so significant. Another buffer need not be supplied immediately. This technique will generally be slightly less effective than the use of longer buffers

because part of each added buffer will be used for QTAM's buffer prefix. Not all of each added buffer will be available for data. In addition, the use of more, but shorter, buffers will mean less efficient use of QTAM queue space on disk.

9) User programs, both foreground and background, should make minimal use of the console typewriter as an output device in order to not generate additional conflicts for the typewriter. Use of the console as an input device for user programs should be prohibited. In a system which has a Teleprocessing operating in a multiprogrammed environment, the console should not be locked up waiting for operator response to a user program message.

10) Foreground initiation of other programs while a Teleprocessing program is running should minimize use of the console typewriter in order to reduce typewriter conflicts. This implies that a card reader should be assigned for use in foreground initiation if it is to be done with any frequency.

11) PAUSE cards should not appear in the background job stream when a Teleprocessing foreground program is running. The use of PAUSE locks up the console waiting for a (B) response from the operator. In place of PAUSE, use a comments card followed by a STOP BG card. The background job stream can then be restarted by typing in 'START BG'.

DOS/360 RPG ACHIEVING TWO AREA OVERLAP ADDITION

The following is a field contribution which has not been submitted to any formal IBM test. The use of this technique should be evaluated in the intended environment prior to implementation.

Newsletter 67-16, dated August 11, 1967, contained, on page 3, a technique for achieving two-area overlap in DOS/360 RPG. The coding for the card reader would not work when a table file and data file were being read from cards because only the first DTF, the one for the table file, was being modified. The coding necessary to allow operation with the table file and data file being read from cards is shown in Exhibit 1.

By testing the first time switch in the DTF, it is determined whether this DTF has been modified. By using a switch to keep track of which IO area is being utilized, after the first file it can be determined which address should be loaded into the CCW. If it is not the first file, the first time switch is turned off so that the logic module will not issue two reads.

S/360-20 DISK RPG MULTIPLE CHAINING TO THE SAME FILE

The following is a field contribution.

1. The S/360-20 Disk RPG SRL's indicate that you cannot chain to a file more than once from the same record, or from a file to itself, in S/360-20 Disk RPG.

2. The chaining described above can be accomplished by defining the chained file more than once in the RPG program, each time with a different file name.

3. Additional VOL, DLAB, and XTENT control cards are required for successive definitions of the file. They are identical to those normally used for the file, with the exception of the file name in the VOL card.

4. Updating a multiple-chained file as a result of each chaining can lead to erroneous results and should be avoided.

DOS/360 RPG ACHIEVING TWO AREA OVERLAP ADDITION

```
IJCFZIZ3    START    0                          NAME OF START = RPG CDMOD
            DS       CL8                        DUMMY 8 BYTES FOR OFFSET
            VALR     5,0
            USING    *, 5
            STM      0, 15, SAVE                SAVE REGISTERS
            TM       21 (1), X '40'             DTF FIRST TIME SWITCH
            BC       1, ONE 1                   NOT FIRST TIME FOR THIS DTF
            MVC      SAVEWK, 32 (1)             SAVE WORK AREA ADDRESS
            MVC      40 (10, 1), MOVE           MODIFY RPG DTF FOR WORK AREA

ENTER       BC       0, ONE 2                   FOR FIRST DTF MODIFIER
            OI       ENTER + 1, X 'FO'          CHANGE TO BRANCH FOR ADDITIONAL DTFS

ENT1        MVC      25 (3,1), ADDCOND          INITIALIZE DTF I/O AREA 2
            MVC      33 (3,1), IOADDC           CHANGE CCW ADDRESS TO IO AREA 1
ONE1        L        0, SAVEWK                  LOAD ADDRESS OF WORK AREA
            XI       SWITCH, X 'FF'             KEEP TRACK OF WHICH IO AREA USED
            L        15, = A (LGMDCRPG)         SET UP FOR BRANCH TO LOGIC MOD
            BAL      14, 8 (15)                 GO TO LOGIC MODULE
            MVC      SAVE + 4 (4), 24 (1)       RETURN FROM LOGIC MODULE
            LM       0, 15, SAVE                RESTORE REGISTERS
            BR       14                         RETURN TO RPG
ONE2        OI       21 (1), X '40!             TURN OFF DTF FIRST TIME SWITCH
            CLI      SWITCH, X '00'             CHECK WHICH IO AREA LAST USED
            BE       ENT1                       READY TO USE IO AREA 1
            MVC      25 (3,1) IOADDC            INITIALIZE DTF IO AREA 2 WITH ADDR IO AREA 1
            MVC      33 (3,1), ADDCOND          CHANGE CCW ADDR TO IO AREA 2
            B        ONE1
            DS       OD
SAVE        DS       16F                        REGISTER SAVE AREA
INAREA      DS       CL80                       IO AREA11
INPUT2      DS       CL80                       IO AREA 2
IOADDC      DC       AL3 (INAREA)
MOVE        DC       X'47000000D24FD000E000'    NOP,MVC
SWITCH      DC       X'00'
ADDCOND     DC       AL3 (INPUT2)
SAVEWK      DS       F                          WORKAREA ADDRESS SAVE AREA
            EXTRAN   LGMDCRPG                   LOGIC MODULE
            END      IJCFZIZ3


LGMDCRPG    CDMOD    DEVICE=2501,WORKA=YES,IOAREA2=YES,SEPASMB=YES
            END
```

Exhibit 1

S/360-44 USE OF 2415 UNITS

Inquiries have been made regarding the use of 2415 Magnetic Tape Units with the S/360-44 Programming System. The 2415 Tape Units are available as a no charge RPQ, but are not officially supported under the S/360-44 Programming System.

This article is intended to clarify the use of 2415's with S/360-44. The primary area of consideration is that of the differences between the 2415 and 2400 hardware error detection. The 2400 software support is considered to be suitable for use with the 2415, but no official IBM testing has been conducted with the 2415.

The 2415-2400 basic differences affect data check handling on a read operation as follows:

1) The 2415 hardware writes a CRC (Cyclic Redundance Check) after each record on to Model 1-2-3 tapes, but the CRC byte is ignored in read operations. The request TIE (Track In Error) mode set is therefore treated as a no-op and there is no automatic hardware recovery for a single track in error. Nevertheless, the read operation will be retried the standard amount of times. The procedure is then identical to a multiple track in error recovery for the 2400. Automatic in-flight single track error correction is provided for the 2415 Models 4-5-6 as for 2400 Models 4-5-6.

2) Noise Bit: On 2400 Models 1-2-3, data check is raised upon detection of VRC (Vertical Redunancy Check), LRC (Longitudinal Redundancy Check), SKEW, SKEW VRC or CRC (Cyclic Redundancy Check). When end of data is detected while reading from a tape, LRC and CRC are checked and data check possibly raised. End of data is detected at the end of the record, end of noise record, or in the middle of a record if data is masked out by unclean tape (broken record). The noise bit is used to differentiate between noise record and "broken record". If data is detected after end of data, it is a "broken record" and the noise bit is turned on. This is useful if the record is "broken" within the first 12 bytes, as the number of bytes transmitted indicates noise length record. On a 2415, the noise bit is not supported by the hardware, so there is no way to differentiate between a noise record and a record "broken" within the first 12 bytes. This would cause records to be lost.

   This problem does not exist with 2400 models 4-5-6 and 2415 models 4-5-6 in phase encoded mode, as phase encoded records begin with a preamble of 41 bytes and finish with a postamble of 41 bytes. Data transmission is activated after detection of the preamble and noise records are not read in.

TELEPROCESSING - 303 DATA STATION OPTIONS REDEFINED

As part of a previous Newsletter article (Issue 67-23), the latest set of options for these data stations was given. Further discussions with the common carriers has produced a list which essentially does not change the operation, but which describes the options in the language used in the carriers current field literature.

Remember that there are essentially three kinds of options available on data sets:

1. Installer options which can only be decided by the installer based on the line and other characteristics.

2. Customer-machine options which are dictated by the design of the machine.

3. Customer options dictated strictly by his desired mode of operation.

Those options which IBM must insure are properly specified are the second class-customer-machine options. These are listed below for the Western Electric 303 data stations (Suffix B, C, D,).

1. Unbalanced Customer Interface
2. With Scrambler-Descrambler
3. Internal Transmitter Clock (Z)
4. Free Running Scrambler (Q)
5. Synch Logic Normal (J)

In addition, it is important to note that we do not want Permanent Request to Send (T) to be specified. Also, the customer will have to make a choice between no voice, alternate voice, or simultaneous voice for coordinating purposes.

The use of the alphabetic letters on some of the options above is recommended only for the next few months, as a further check on the installers of these fairly new data sets. We continue to be warned by the carriers that such letter designations may change without warning and the phrase designations are much safer for the long term.

NEW PUBLICATIONS ANNOUNCED BY PRL'S

The weekly PRL's (Publications Release Letters) are used to insure that all Salesmen and Systems Engineers are aware of new or revised Marketing Publications. Normally, each issue of the Newsletter will contain information from two PRL's, one following the other. The information will be placed in the Newsletter in its original form with no rearrangement of form numbers or titles. It is not intended to replace existing information and distribution sources. You should be certain that you are aware of these sources.

INFORMATION ABOUT THE NEWSLETTER

EVERY SALESMAN AND SYSTEMS ENGINEER SHOULD RECEIVE ONE COPY OF THE NEWSLETTER. Each FE Branch Manager should receive five copies for distribution to Customer Engineers.

The IBM Installation Newsletter is distributed only to IBM locations and is not available to customers. The Newsletter is separated into two parts. The first part contains information which is intended only for IBM personnel and the pages are labeled for IBM Internal Use Only. THESE PAGES, OR REPRODUCTIONS OF THEM, ARE NOT TO BE GIVEN TO CUSTOMERS. However, using your own good judgement, you may discuss pertinent information from these pages with a customer. The second part contains information which may be REPRODUCED by the Branch Office at their discretion and given to customers. These pages are not labeled with any restrictive classification. Note that when any abstracts for Type III programs are given to the customer, he must be informed that the disclaimer for Type III programs, as contained in the Newsletter, applies. YOUR COOPERATION IN PROPERLY HANDLING THE CONTENT OF THE IBM INSTALLATION NEWSLETTER IS NECESSARY FOR ITS CONTINUED PUBLICATION.

Changes to the quantity of the Newsletter shipped are to be made by the Administration Manager. (NOTE THAT EXTRA COPIES OF THE NEWSLETTER ARE NOT TO BE ORDERED FOR DISTRIBUTION OF INFORMATION TO CUSTOMERS.) He should update and send the DP Administration Managers' Communication Control card (form number N44-780) to DP Communication Control, DPD HQ.

Direct inquiries concerning the contents of this Newsletter to:

W. P. Gaffigan, Editor
IBM Installation Newsletter
Systems Engineering Technical Programs
DPD HQ
112 East Post Road
White Plains, New York 10601
Telephone 914 WH9-1900 Extension 4781

PRL #50 December 22, 1967

MARKETING PUBLICATIONS

FORM NO.	TITLE	DISPOSITION
250-0045-1	Design and Analysis of Communication Based Systems	Scrap 250-0045-0
420-5047-1	Transaction Simulator System for the IBM 2260 Display Station	Revised Abstract
504-1277-2	Problem Solvers	NEW
520-2013-0	The IBM Computing Accounting Machine: computing power with accounting machine economy	NEW
542-0047-0	IBM Selectric Instruction Book	NEW
A22-6884-2	IBM System/360 Model 65 Functional Characteristics	Use A22-6884-1 & N27-2909
A22-6900-2	Data Acquisition Special Features for the IBM S/360 Model 44	Scrap A22-6900-1
A26-3567-3	IBM 2311 OEMI	Scrap A26-3567-2, -1
A27-2700-2	IBM S/360 Component Description, IBM 2260 Display Station/IBM 2848 Display Control	Use A27-2700-1, N27-2905, N27-2908, N27-2910 & N27-2913
C24-3408-2	IBM S/360 BPS I/O 1231 Specifications and Operating Guide	Scrap C24-3408-1, N24-5073, N24-5205, N24-5072 & N24-5206
C26-3629-1	IBM 1130 Card/Paper Tape Programming System Operator's Guide	Scrap C26-3629-0, N26-0556, N26-0518 & N26-0526
C26-3706-3	IBM 1130 Synchronous Communications Adapter Subroutines	Scrap C26-3706-2
C26-3801-4	IBM S/360 Model 20 Punched Card Utility Programs Operating Procedures	Scrap C26-3801-3, -2 & N33-8501
C27-6909-4	IBM S/360 Operating System Graphic Programming Services for IBM 2250 Display Unit	Scrap C27-6909-3 & N27-1278
H20-0321-2	Attached Support Processor System (ASP) (360A-CX-15X) Operator's Manual	Scrap H20-0321-1
H20-0322-1	Attached Support Processor System (ASP) (360A-CX-15X) Application Programmer's Manual	Scrap H20-0322-0
H20-0477-0	IBM S/360 Document Processing System (360A-CX-12X) Program Description and Operations Manual	NEW
H20-0481-0	Retail IMPACT Fashion System (360A-DR-04X) Operations Manual	NEW
K20-0237-0	Direct Access Personal Trust Accounting First Wisconsin National Bank and First Wisconsin Trust Co.	NEW
N24-5350	TNL to IBM S/360 BPS I/O 1418/1428 Specifications and Operating Guide Re: C24-3437-2	NEW
N26-0571	IBM 1800 Time-Sharing Executive System Operating Procedures Re: C26-3754-2 with N26-0561, N26-0569	NEW
N28-0240	IBM S/360 Disk & Tape Operating Systems COBOL Language Specifications Re: C24-3433-3, -4 & N24-5188, N28-0232, N28-0237	NEW
R20-1061-0	IBM S/360 QTAM Learner's Guide I	NEW
R20-9206-0	IBM 1130/1800 FORTRAN IV Workshop Course Description	NEW
R20-9150-3	IBM S/360 COBOL Fund. P.I. Course Description	Use R20-9150-2
R20-9151-3	IBM S/360 COBOL P.I. Course Description	Scrap R20-9151-2
R20-9152-3	IBM S/360 P.I. COBOL Course Description	Scrap R20-9152-2
R20-9217-1	IBM S/360 DOS/TOS Language Interface	Scrap R20-9217-0
R29-0082-4	IBM S/360 FORTRAN P.I. Chapter 2	Use R29-0082-3
Y20-0122-0	Integrated Planning and Control System for the Making and Continuous Casting of Steel	NEW
Y26-3620-1	IBM 1130/1800 Card/Paper Tape Programming Systems Program Logic Manual	Scrap Y26-3620-0, Y26-0507, Y26-0504, Y26-0524 & Y26-8003
Y26-3714-0	IBM 1130 Disk Monitor Programming System, Version 2, PLM	NEW
Y26-8005-0	TNL IBM 1130 Monitor Programming System PLM Re: Y26-3752-0	NEW
Y26-8008-0	IBM 1800 Time-Sharing Executive System, PLM Re: Y26-3702-1	NEW
Z77-7237-0	The 1130 for Wholesale Grocery Accounting	NEW
Z77-7271-0	A S/360 RPG Square Root Subroutine	NEW
Z77-7278-0	Assembler and COBOL Program Linkage	NEW
Z77-7279-0	An Approach to Minimizing Operator Control in an OS/360 MFT Environment	NEW
Z77-7280-0	Index Sequential Access Method Coding for COBOL E Under OS	NEW
Z77-7281-0	SRI-The Inter Digital Method for Random Number Generation	NEW

PRL #51 (December 22, 1967

A26-5916-2	IBM 1130 Bibliography	Scrap A26-5916-1
N20-0030-16	Catalog of Programs for IBM S/360 - November 1967 Supplement to C20-1619-3	Scrap N20-0030-15
N20-0031-12	Catalog of Programs for IBM 1130 Computer System and IBM 1800 Data Acquisition November 1967 Supplement to C20-1630-2	Scrap N20-0031-11
N20-0360.44	S/360 SRL Newsletter Re: A22-6822-10	Scrap N20-0360.43

MARKETING PUBLICATIONS

221-0351-3	The IBM 1050 Data Communication System Proposal Insert	Use 221-0351-2
221-0401-2	The IBM 2203 Printer Proposal Insert	Use 221-0401-1
221-0405-2	The IBM 2560 Multi-Function Card Machine Proposal Insert	Use 221-0405-1
520-1519-0	Binary Synchronous Communications Promotional Brochure	NEW
545-0054-1	224 Spec Sheet	Use 545-0054-0
A19-0010-3	3944 Dial Terminal Unit	Scrap A21-0010-2
A24-3045-4	IBM 1030 Configurator	Scrap A24-3045-3
A24-3231-4	S/360 Functional Characteristics	Use A24-3231-3
A24-3475-4	402-407 Computing Accounting Machine	Use A24-3475 & N24-0408
A27-2701-1	IBM S/360 Component Description IBM 2250 Display Unit Model 1	Use A27-2701-0 & N27-2904
A27-2728-0	IBM S/360 Model 65 Operator's Guide	NEW
C20-1666-1	Data Communications Glossary	Use C20-1666-0
C22-6796-1	IBM 7740 Installation Manual - Physical Planning	Use C22-6796-0, N22-0138, N22-0188
C24-3322-3	FORTRAN IV Language Specs. IBM 1401, 1440, 1460 Programming Specifications	Use C24-5322-2, N21-5051 & N21-5061
C24-3355-5	IBM S/360 BPS Assembler with I/O Macros Specifications	Scrap C24-3355-4, N24-5112, N24-5144, N24-5118, N24-5167 & N24-5313
C24-3391-3	IBM S/360 BPS BTOS Guide	Scrap C24-3391-1, -2, N24-5083, N24-5115 & N24-5162 & N24-5177
C24-3409-3	IBM S/360 BOS Utility Programs Specifications	Scrap C24-3409-2 & C24-3455-1
C24-3430-3	IBM S/360 TOS Data Management Concepts	Scrap C24-3430-2 & N24-5194
C24-5015-5	IBM S/360 TOS System Gen. & Maintenance	Scrap C24-5015-4
C24-5020-4	IBM S/360 TOS Performance Estimates	Scrap C24-5020-3 & N24-5267
C24-5021-3	IBM S/360 TOS Operating Guide	Scrap C24-5021-2, N24-5209, N24-5288 & N24-5300
C24-5035-3	IBM S/360 TOS Supervisor and I/O Macros	Scrap C24-5035-2 & N24-5273
C24-5036-2	IBM S/360 DOS System Control and System Service Programs	Use C24-5036-1, N24-5218, N24-5279, N24-5330, N24-5248 & N24-5302
C24-5066-1	IBM S/360 DOS & TOS Exec. Specifications Operator's Guide	Scrap C24-5066-0
C26-3605-2	Mod 20 Sterling Currency Processing Routines	Scrap C26-3605-1 & N24-9001
C26-3717-0	IBM 1130 Disk Monitor System Version 2 Programming and Operator's Guide	NEW
E20-0297-0	System Application Concept	NEW
E20-8033-1	General Ledger and Financial Control	Use E20-8033-0
H20-0480-0	Retail IMPACT Fashion System (360A-DR-04X) Program Description Manual	NEW
J20-0021-0	Industry Information Service - Sales Manual	NEW

PRL #51 December 22, 1967

K20-0244-0	The Production and Inventory Control System at the Kraft Division, St. Regis Paper Company with the IBM S/360	NEW
N20-1061	TNL to System/360 Remote Access Computing System (RAX) (360A-CX-17X) Application Description Re: H20-0353-0	NEW
N20-1062	TNL to S/360 Remote Access Computing System (RAX) (360A-CX-17X) Operator's Manual Re: H20-0355-0	NEW
N20-1063	TNL to Structural Engineering System Solver (STRESS) for the IBM 1130, Model 2B (1130-EC-03X) Version 2, User's Manual Re: H20-0340-2	NEW
N21-5055	IBM S/360 Disk & Tape Operating System RPG Specifications Re: C26-3570-4	NEW
N21-5074	TOS Autotest Specifications TNL Re: C24-3441-3	NEW
N21-5080	IBM S/360 TOS Autotest Specifications Re: C24-3441-3 with N21-5054 & N21-5074	NEW
N23-0606	ENL IBM S/360 Mod 20 RPG P.I. Advisor Guide Re: R29-0090-1 with N23-0609	NEW
N24-5236	TNL to IBM S/360 BPS System Generation and Maintenance Re: C24-5061-1	NEW
N24-5325	TNL to IBM S/360 BOS Operating Messages Re: C24-5024-1 & -2	NEW
N24-5335	TNL to BOS Assembler in I/O Macros Specifications Re: C24-3361-5 with N24-5314	NEW
N24-5336	TNL to S/360, BOS Programmer's Guide Re: C24-3372-6	NEW
N24-5338	TNL to S/360 BOS System Generation & Maintenance Re: C24-5060-1 with N24-5261, N24-5315 & N24-5332	NEW
N24-5339	TNL to IBM S/360 BPS BTS Programmer's Guide Re: C24-3354-6	NEW
N24-5348	TNL to S/360 TOS System Control & System Service Progs. Re: C24-5034-1	NEW - Scrap N24-5271
N28-3006	IBM S/360 Time Sharing System Assembler User Macro Instructions Re: C28-2004-1	NEW
R20-1014-3	IBM S/360 RPG Coding Practice Problems	Use R20-1014-2
R29-0081-4	FORTRAN IV for IBM S/360 P.I. Chapter 1	Use R29-0081-3
R29-0083-4	FORTRAN IV for IBM S/360 P.I. Chapter 3	Use R29-0083-3
X20-1727-2	IBM S/360 Model 20 Reference Data	Scrap X20-1727-1
X20-1744-2	PL/I Reference Data Card	Scrap X20-1744-1
X20-1745-1	QUIKTRAN Reference Card	Scrap X20-1745-0
X20-1746-3	OS/360 JCL Reference Card	Scrap X20-1746-2
Y20-0136-0	IBM Punched Card Data Processing for Property and Liability Insurance	NEW
Y28-3054-0	IBM S/360 Time Sharing System Program Checkout Subsystem PLM Re: Y28-2014-0	NEW
Y33-9008-0	IBM S/360 Mod 20 Disk Programming System Control and Service Programs PLM	NEW
Z32-0393-0	ENL IBM Comp. System Training Student Handouts (Z32-0400-1)	NEW
Z32-0395-0	ENL IBM CST Education Guide Part II Re: Z32-0402-1 with Z32-0399	NEW
Z77-7283-0	Data Management Debug Sheet	NEW
Z77-7287-0	Tips on Installing 1050 Terminals	NEW

PRL #1 (2) January 5, 1968

MARKETING PUBLICATIONS

221-0459-0	The IBM 2065 Processing Unit Proposal Insert	NEW
221-0460-0	The IBM 2075 Processing Unit Proposal Insert	NEW
221-0478-0	The IBM 2025 Processing Unit Proposal Insert	NEW
221-0479-0	The IBM 633 Billing System Proposal Insert	NEW
320-0950-0	Standard Industrial Classification Manual, 1967 Edition	NEW
420-0174-0	IBM 633 Billing System	NEW
520-1912-0	IBM S/360 Model 20: for the property & liability insurance company	NEW
520-1932-0	Linear Programming for Management Decisions Promotional Brochure	NEW
520-2037-0	IBM S/360 Model 25: The New Intermediate Computing System for Pertoleum Marketing	NEW
520-2038-0	IBM S/360 Model 25: The New Intermediate Computing System for Utilities	NEW
520-2039-0	IBM S/360 Model 25: The New Intermediate Computing System for Science, Engineering, and Management Science	NEW
520-2040-0	IBM S/360 Model 25: The New Intermediate Computing System for Distribution Industries	NEW
520-2041-0	IBM S/360 Model 25: The New Intermediate Computing System for Colleges and Universitites	NEW
520-2042-0	IBM S/360 Model 25: The New Intermediate Computing System for the Financial Industry	NEW
520-2043-0	IBM S/360 Model 25: The New Intermediate Computing System for the Federal Government	NEW
520-2044-0	IBM S/360 Model 25: The New Intermediate Computing System for Life Insurance Companies	NEW

PRL # 1 (2) Continued

520-2045-0	IBM S/360 Model 25: The New Intermediate Computing System for Property and Liability Insurance Companies	NEW
520-2046-0	IBM S/360 Model 25: The New Intermediate Computing System for Hospitals	NEW
520-2047-0	IBM S/360 Model 25: The New Intermediate Computing System for Manufacturing Industries	NEW
520-2048-0	IBM S/360 Model 25: The New Intermediate Computing System for the Paint Industries	NEW
520-2049-0	IBM S/360 Model 25: The New Intermediate Computing System for Foundries	NEW
520-2050-0	IBM S/360 Model 25: The New Intermediate Computing System for State and Local Government	NEW
520-2051-0	IBM S/360 Model 25: The New Intermediate Computing System for the Communication Industry	NEW
520-2052-0	IBM S/360 Model 25: The New Intermediate Computing System	NEW
520-2056-0	IBM Model 65 Multiprocessing System Promotional Brochure	NEW
520-2062-0	The IBM 633 Billing System Promotional Brochure	NEW
520-2064-0	IBM S/360 Model 25 Facts Folder	NEW
A24-3510-0	IBM S/360 Model 25 Functional Characteristics	NEW
A24-3511-1	IBM S/360 Model 25 Cinfigurator	Scrap A24-3511-0
A24-3512-0	IBM S/360 Model 25 1401/1460 & 1440 Compatibility Features	NEW
C26-3602-4	IBM S/360 Model 20 Basic Assembler Language (Card & Tape)	Scrap C26-3602-3 & N33-8511
C28-2033-0	IBM S/360 Time Sharing System Operator's Guide	Scrap C28-2033-0
J20-0020-0	Industry Information Service - Client's Manual	NEW
K20-0240-0	Transit Processing on the IBM S/360 with Direct Access Storage at American National Bank & Trust Company	NEW
N20-1064	TNL for System/360 AUTOSPOT Numerical Control Processor (360A-CN-08X) Version 2 - Application Description Re: H20-0462-0	NEW
N20-1068	TNL for Retail IMPACT - Application Description Re: E20-0188-2	NEW
N21-0089	IBM 1231-N1 TNL to A21-9031-0 with N21-0068 & N21-0061-1	NEW
N21-0090	IBM 1442-N1 & N2 TNL Re: A21-9025-3 with N21-0082	NEW
N21-5087	BPS RPG (Tape) Specifications Re: C24-3418-1 & -2 with -1 , N24-5056, N24-5160, N24-5176, N24-5200 with -1 & -2, N21-5059	NEW
N22-0286	Changes to IBM S/360 System Summary Re: A22-6810-8 with N22-0276	NEW
N22-0288	Changes to System/360 Installation Manual Physical Planning Re: C22-6820-7	NEW

PRL #1

N23-0127	ENL to IBM Admin. Group Orientation Guide Re: R20-4092-0	NEW
N27-3020	Changes to 7772 Audio Response Unit Component Description Re: A27-2711-0	NEW
N33-1507	TNL to IBM S/360 Model 20 Functional Characteristics Re: A26-5847-2, -3 with N26-0174, N26-0176, N26-0179, N33-1501 & N33-1504	NEW
R20-1060-1	IBM S/360 DOS/TOS Language Interface Class Exercises	Use R20-1060-0
R20-8055-0	IBM S/360 Mod 20 ALC Workshop Education Guide	NEW
R20-9221-0	IBM Branch Office Library Course Description	NEW
R29-0250-0	IBM S/360 Mod 20 ALC Advisor Guide Supplement	NEW
R29-0251-0	IBM S/360 Mod 20 Assembler Language Coding System Review Text	NEW
R29-0252-0	IBM S/360 ALC P.I. Instruction Text	NEW
R29-0253-0	IBM S/360 Mod 20 ALC P.I. Sample Problems	NEW
R29-0254-0	IBM S/360 Mod 20 ALC P.I. Examination	NEW
V20-0092-0	Graphic Data Processing	NEW
V20-0157-0	Executive Introduction to Simulation and GPSS	NEW
V20-0167-0	Introduction to the IBM 2250 Display Unit	NEW
Y20-0143-0	Document Processing System/360 (360A-CX-12X) System Manual	NEW
Y20-0150-0	TNL for System/360 AD-APT/AUTOSPOT Numerical Control Processor (360A-CN-09X) Version 2 System Manual Re: Y20-0118-0	NEW
Y20-0152-0	TNL for 1130 Route Accounting System - System Manual Re: Y20-0078-0	NEW
Y28-3056-0	IBM S/360 Time Sharing System System Service Routines PLM Re: Y28-2018-0	NEW
Z20-1812-1	IBM S/360 Teleprocessing Analysis and Design Program Application Description	Scrap Z20-1812-0
Z20-1866-0	IBM S/360 Model 25 Preliminary Marketing Guide	NEW
Z20-1868-0	TNL to S/360 Model 25 Preliminary Marketing Guide Re: Z20-1866-0	NEW

January 26, 1968 Issue No. 68-01

IBM INSTALLATION NEWSLETTER
TABLE OF CONTENTS

OS/360 RELEASE 13 - DISPLAYING TAPE UNITS INSTEAD OF U/R DEVICES

The following is a field contribution. It consists of a SUPERZAP change made by a user to OS/360 Release 13, SYSGEN'ed for MFT and tested by the user with a S/360 Model 50 in a production environment. Potential users should be aware that it has not been submitted to any formal IBM test, that no maintenance is available, and that they are expected to evaluate its usefulness in their own environment.

The following module change, Job D99755A7, Exhibit 1, is to the display jobnames routine. This causes tape units instead of unit record devices to be displayed on the system console at step initiation time. This gives the operator time to prepare to externally label tapes instead of having to wait until step termination to learn the output tape locations. This has proven more useable to the user than the "Display Status" technique.

OS/360 RELEASE 13 SYSOUT WRITER BYPASS OF TAPE RETENTION

The following is a field contribution. It consists of a SUPERZAP change made by a user to OS/360 Release 13, SYSGEN'ed for MFT and tested by the user with a S/360 Model 50 in a production environment. Potential users should be aware that it has not been submitted to any formal IBM test, that no maintenance is available, and that they are expected to evaluate its usefulness in their own environment.

The following technique overcomes a problem in writing SYSOUT tapes when the label retention period field has been inadvertently modified in the prior use of a labeled scratch tape. The routine permits the SYSOUT writer to ignore the retention period check and saves the operator the trouble of remounting a validly labeled tape.

Job D99755C8, shown in Exhibit 1, overlays a 613 ABEND instruction by branching to user coding which modifies the retention period check routine message and branches to the retention check routine. The message modification code overlays the tape recording technique routine. This was done since this user has no seven track tapes on his system which would require use of the TRTCH routine. If a user had seven track tapes on his system, he would have to find someplace else for the code or reassemble the module with the added coding.

OS/360 RELEASE 13 - DISPLAYING TAPE UNITS INSTEAD OF U/R DEVICES

```
//D99755A7  JOB  90951T,      ,MSGLEVEL=1
// EXEC PGM=SUPERZAP
//SYSLIB   DD  DSNAME=SYS1.LINKLIB,DISP=OLD,VOLUME=SER=A,UNIT=2311
//SYSPRINT DD      SYSOUT=A
//SYSIN    DD      *
*  THIS ZAP CAUSES TAPE DD NAMES TO BE DISPLAYED ON CONSOLE INSTEAD OF
*        UNIT RECORD DD NAMES
          NAME     IEFCNTRL  XTTPCC
          VERIFY   00BE      91086C12
          REP      00BE      91806012
/*
```

Exhibit 1

OS/360 RELEASE 13 SYSOUT WRITER BYPASS OF TAPE RETENTION

```
//D99755C8  JOB  90951T,)      ,MSGLEVEL=1
// EXEC PGM=SUPERZAP
//SYSLIB   DD  DSNAME=SYS1.SVCLIB,DISP=OLD,UNIT=2311,VOLUME=SER=A
//SYSPRINT  DD    SYSOUT=A
//SYSIN    DD    *
*   THIS CHANGE INTERCEPTS A 613 ON AN OUTPUT TAPE.  PUTS A 'Q' IN
*         THE RETENTION PERIOD CHECK MESSAGE AND BRANCHES TO IT.
*  THIS FIX CANNOT BE USED IN A 7 TRACK TAPE INSTALLATION.
          NAME        IGG0190P  IGGC19OP
          VERIFY      022C      43B040C1
          VERIFY      0230      88B00C0442B0
          VERIFY      0236      4022DC00
          VERIFY      023A      4C2233DB
          VERIFY      03B0      41100613
*   REPLACE ABEND.    BRANCH TO PATCH AREA
          REP         03B0      47F0322E
*  BRANCH AROUND TRTCH ROUTINE.  AREA WILL BE USED FOR PATCH.
          REP         022C      47F03248
          REP         0230      D2144C0C33CE
          REP         0236      92D84014
          REP         023A      47F0333E
/*
```

Exhibit 1

OS/360 PASSED DATA SET CLARIFICATION

The following article was supplied by a Field Systems Center.

A recent problem encountered with a passed data set on 7-track tape indicates a need for clarification of how the passing of data sets works. The information in the OS/360 manuals is augmented as follows.

The passed data set results in the filling in of only 6 fields in the JFCB (JFCBDSNM, JFCBELNM, JFCBLTYP, JFCBLFLSQ, JFCBNVOL, and JFCBVOLS) in the receiving step. Examination of these fields indicates that not much information is available - essentially data set name, volume serial numbers, location on volume, and type of label. There are no data set attributes and the tape density field is not supplied.

Therefore, if required, this information must be supplied in the receiving step. With disk data sets and standard labeled tapes, the data set attributes are available in the labels. With nonstandard labels and unlabeled tapes, the data set attributes must be supplied in the program, or the DD card. The tape density field is important in the case of 7-track or 800-1600 BPI 9-track drives. The default for this field is 200 BPI on 7-track and 1600 BPI on the 9-track with 800-1600 BPI capability. Therefore, the system will read the tape at the default density unless a density parameter is supplied on the receiving step.

For example, if an 800 BPI tape with standard labels were created and passed on, a 7-track drive on STEP 1 of a job and STEP 2 referenced it by DSNAME = *. STEP 1. DD1 and DISP=OLD and no density, then the system would default to 200 BPI for density, try to read the tape labels and ABEND with an OPEN problem. STEP 2 must supply the density parameter to avoid the default and to read the labels where the remainder of the information for the data set is found.

OS/360 SIMPLIFIED JCL SYMBOLIC PARAMETER OVERRIDE PROCEDURE

The following article provides additional detail on the new JCL Symbolic Parameter Capability which has been announced (December 11, 1967)

as becoming available February 15, 1968. The information is provided for planning purposes.

Using JCL to override cataloged procedures requires that the problem programmer be familiar with the internal structure of the procedure e.g., he must know the sequence and names of steps and DD statements in order to change any of the parameters. A new facility, symbolic parameters, simplifies procedure invocation by permitting easy and direct substitution for any parameter on any statement of a cataloged procedure. In addition, symbolic parameter substitution allows problem programmers to override cataloged procedures using non-JCL oriented statements.

Symbolic parameters, which are syntactically analogous to assembly language macro parameters, are described in the publication "IBM System/360 Operating System-Job Control Language", Form C28-6539-4. Briefly, symbolic parameters can be defined in a cataloged procedure using the special character "&", and values may subsequently be assigned to these parameters on EXEC statements invoking the procedure. A new JCL statement, PROC, allows default values to be specified within the procedure.

An example of the type of procedure which could use this facility is an assembly and linkage edit, where the source is a member of PDS library. Some of the input parameters to a procedure of this type might be:

1. MODULE to be processed.
2. TYPE of module (e.g., reentrant).
3. Private macro library (MACLIB) to be used.
4. Volume (MACVOL) containing the private macro library.

Exhibit 1 shows the current cataloged procedure and an example of modification to use symbolic parameters.

Note that the underlined parameters are the symbolic parameters which are to be overridden. Also, as in the case of & MODULE, they can be used more than once. The PROC statement defines a null value as the default for &TYPE. (This defaults to a "safe" attribute of non-reusable unless a less restrictive attribute is explicitly assigned.)

The JCL which a problem programmer would need to override the current cataloged procedure is shown in Exhibit 2.

Note the procedure-dependent information which the problem programmer must supply:

1. Procedure stepnames must be known.
2. DDnames within steps of the procedure must be known.
3. The order in which steps appear within the procedure must be known.
4. DDnames must be overridden in the same order that they appear in the step within the procedure.
5. DD's must be shown for unmodified members of a concatenation, (but only if they precede the modified members).
6. All processor options must be stated even if they aren't being changed.
7. Overrides must contain JCL information which is not modified if it is contained in the same major key field as a modification.

The symbolic override consists of only one statement:

//EXEC ASMED, MODULE = MYMOD, TYPE= RENT, MACLIB=PVTMAC, MACVOL=792613.

Note that the keywords in this statement permit the user to describe his job in a mnemonic problem-oriented way and that he is almost entirely shielded from the JCL.

OS/360 ISAM - IMPROVEMENT OF DATA SET INTEGRITY

Improvement of data set integrity can be accomplished under OS/360 ISAM.

PTF # 360S-13270-12 is available to correct the problem described below. This PTF may be used with Releases 12, 13, and 14, and supersedes PTFs 10230-14 and 11341-11. The following APARs have been fixed by this PTF: 13270, 10230, and 11341. Inquiries regarding these PTFs and APAR's should be directed to

OS/360 SIMPLIFIED JCL SYMBOLIC PARAMETER OVERRIDE PROCEDURE

The current cataloged procedure to do the assembly and linkage edit would be:

```
//ASM        EXEC   PGM=IEUASM
//SYSLIB     DD     DSNAME=SYS1.MACLIB,DISP=OLD
//           DD     DSNAME=SOMEMAC,VOLUME=(PRIVATE,SER=123456),DISP=OLD
//SYSIN      DD     DSNAME=SYMLIB(SOMEMOD)
//SYSUT1     DD     UNIT=SYSSQ,SPACE=(1700,(400,50))
//SYSUT2     DD     UNIT=SYSSQ,SPACE=(1700,(400,50))
//SYSUT3     DD     UNIT=(SYSSQ,SEP=SYSUT2,SYSUT1,SYSLIB)),                  x
//                  SPACE=(1700,(400,50))
//SYSPRINT   DD     SYSOUT=A
//SYSPUNCH   DD     DSNAME=&LOADSET,UNIT=SYSSQ,SPACE=(80,(200,50)),          x
//                  DISP=(MOD,PASS)
//LKED       EXEC   PGM=IEWL,PARM='XREF,LIST,NCAL'
//SYSLMOD    DD     DSNAME=&TEMP(PDS),UNIT=SYSDA,SPACE=(1024,(50,20,1))
//SYSLIN     DD     DSNAME=&LOADSET,DISP=(OLD,DELETE)
//           DD     DDNAME=SYSIN
//SYSUT1     DD     UNIT=(SYSDA,SEP=(SYSLIN,SYSLMOD)),SPACE=(1024,(50,20))
//SYSPRINT   DD     SYSOUT=A
```

If this procedure were modified to use symbolic parameters, it might appear as shown below.

```
           PROC   TYPE=
//ASM      EXEC   PGM=IEUASM
//SYSLIB   DD     DSNAME=SYS1.MACLIB,DISP=OLD
//         DD     DSNAME=&MACLIB,VOLUME=(PRIVATE,SER=&MACVOL),DISP=OLD
//SYSIN    DD     DSNAME=SYMLIB(&MODULE)
//SYSUT1   DD     UNIT=SYSSQ,SPACE=(1700,(400,50))
//SYSUT2   DD     UNIT=SYSSQ,SPACE=(1700,(400,50))
//SYSUT3   DD     UNIT=(SYSSQ,SEP=(SYSUT2,SYSUT1,SYSLIB)),                  x
//                SPACE=(1700,(400,50))
//SYSPRINT DD     SYSOUT=A
//SYSPUNCH DD     DSNAME=&LOADSET,UNIT=SYSSQ,SPACE=(80,(200,50)),           x
//                DISP=(MOD,PASS)
//LKED     EXEC   PGM=IEWL,PARM='XREF,LIST,NCAL,&TYPE'
//SYSLMOD  DD     DSNAME=OBLIB(&MODULE),UNIT=SYSDA,SPACE=(1024,(50,20,1))
//SYSLIN   DD     DSNAME=&LOADSET,DISP=(OLD,DELETE)
//         DD     DDNAME=SYSIN
//SYSUT1   DD     UNIT=(SYSDA,SEP=(SYSLIN,SYSLMOD)),SPACE=(1024,(50,20))
//SYSPRINT DD     SYSOUT=A
```

Exhibit 1

OS/360 SIMPLIFIED JCL SYMBOLIC PARAMETER OVERRIDE PROCEDURE

```
                                  (1)    (6)   (6)   (6)
   //EXEC       ASMED, PARM.LKED='XREF, LIST, NCAL, RENT'
                   (2)(4)  (5)
(3)//ASM.SYSLIB      DD
                                                     (7)
   //                DD     DSNAME=PVTMAC, VOLUME=(PRIVATE, SER=792613)
                 (2) (4)
   //ASM.SYSIN       DD     DSNAME=SYMLIB(MYMOD)
                     (2)
(3)//LKED.SYSLMOD    DD     DSNAME=OBLIB(MYMOD)
```

Exhibit 2

the local Field Engineering representative.

. Problem

After a WRITE-KEY-NEW operation, the pointers in the Format 2 DSCB which point to the last record in the prime area (DS2-LPRAD) and to the last record in the independent overflow area (DS2LOVAD) may be invalid. At CLOSE time, the last prime data record pointer (DCBLPDA) and the last independent overflow record pointer (DCBLIOV) fields of the DCB are used to update the format 2 DSCB. If the ISAM data set is not successfully closed, the format 2 DSCB may not accurately describe the structure of the ISAM data set.

. Correction

This problem can be solved by maintaining an End of File (EOF) record just beyond the last independent overflow record and checking the location of this EOF record as well as the existing EOF record at the end of the prime data upon opening the data set. If necessary, the DCB fields which reflect these locations are corrected. This check of critical fields at OPEN time will improve data set integrity.

. Programming Considerations

If ISAM with PTF is used to process an existing data set, any residual EOF record in the independent overflow area will cause the corresponding last record pointer to be erroneously adjusted. Space between the actual last record and the residual EOF record will become unavailable. It is suggested that the data set be reorganized using ISAM with PTF code.

When calculating space requirements for an indexed sequential data set, the user must consider the space required for the EOF record following the last data record in the prime and independent overflow areas. ISAM reserves the last track of the prime area for the EOF record. In the independent overflow area the space required for the EOF record is equal to the BLKSIZE (block size). Therefore, when determining the number of records that will fit in the independent overflow area, the EOF record must be considered.

OS/360 RELEASE 13 - COBOL F PERFORMANCE AND DATA SET SPACE IMPROVEMENT

The following is a field contribution. It consists of a SUPERZAP change made by a user to OS/360 Release 13, SYSGEN'ed for MFT and tested by the user with a S/360 Model 50 in a production environment. Potential users should be aware that it has not been submitted to any formal IBM test, that no maintenance is available, and that they are expected to evaluate its usefulness in their own environment.

The following change will improve COBOL F performance and reduce the data set space requirement.

This SUPERZAP change, Job D99755JC, Exhibit 1, changes the blocking of the COBOL F

SYSLIB dataset. The change is to blocksize constant modifying a blocksize from 400 to 3600 giving this user better 2311 disk utilization.

OS/360 RELEASE 13-COBOL F PERFORMANCE AND DATA SET SPACE IMPROVEMENT

```
//D99755JC JOB 90951T,          ,MSGLEVEL=1
// EXEC PGM=SUPERZAP
//SYSLIB   DD  DSNAME=SYS1.LINKLIB,DISP=OLD,VOLUME=SER=A,UNIT=2311
//SYSPRINT DD    SYSOUT=A
//SYSIN    DD    *
*    THIS ZAP ALLOWS BLOCKING OF SYSLIB FOR THE COBOL F COMPILER
         NAME  IEQCBL00      IEQCBL00
       VERIFY   041C    019C
      REP   041C   0E10
/*
```

Exhibit 1

OS/360 FORTRAN - OPM'S TO PUT POUND SIGN BACK INTO CSECT NAME

There are three OPM's (Optional Program Modifications) that will be available with Release 14 in SYS1.SAMPLIB. The application of these OPM's will allow an installation to continue with the CSECT pound sign convention and operate in the same manner as at present. They will be available for a limited period of time, and will allow additional time for a normal conversion.

The first step in applying these OPM's is to list and punch OPM00001 (applies to FORTRAN E), OPM00002 (applies to FORTRAN G) and OPM00003 (applies to FORTRAN H) which are three separate members of SYS1.SAMPLIB. Each of these members is made up of two job steps, the first to update SYS1.MODLIB, and the second to update SYS1.LINKLIB. The OPM's may be used as they are supplied, but for efficiency the following alternate approach is suggested. There are two separate circumstances where these OPM's may be applied. The first is before SYSGEN and the second is after SYSGEN. The advantage of applying them before SYSGEN is that it need only be done once for multiple SYSGENs. The advantage of applying them after SYSGEN is that with only one SYSGEN you can have a system with the OPM's applied and a second system without them. This can be done by copying the new system and applying the OPM's applied and a second system without them.

In either case only one of the two job steps supplied in each OPM need be run.

To make the application before SYSGEN, the following job steps should be to update SYS1.MODLIB for FORTRAN'S E, G & H, respectively:

//STEP1	in OPM00001
//LKEDMOD	in OPM00002
//STEP1	in OPM00003

To make the application after SYSGEN, the following job steps should be run to update SYS1.LINKLIB for FORTRAN'S E & G, respectively:

//STEP1	in OPM00001
//LKEDMOD	in OPM00002

The DD card for SYSLMOD should be changed to point to SYS1.LINKLIB instead of SYS1.MODLIB before either of these job steps is run.

To make the application after SYSGEN, the following job step should be run, without change, to update SYS1.LINKLIB for FORTRAN H:

//STEP2 in OPM00003

Currently, when a module in SYS1.LINKLIB is replaced by a new module, the space used by the old module is not made available for reuse. For this reason the following amounts of unused space must be available in SYS1.LINKLIB in order to successfully apply the OPM's after SYSGEN. FORTRAN E requires 4 additional tracks, FORTRAN G requires 5 additional tracks and FORTRAN H requires 152 additional tracks.

OS/360 PL/I DEBUGGING AID

It is sometimes desirable, when debugging large PL/I programs, to be able to access the number of the statement causing a particular error. This normally occurs when an ON unit is written such that its scope is for the duration of the program, and that execution must be forced as far as possible in order to find the maximum number of bugs per run, and when the standard system action (which would automatically print the statement number) is not suitable.

The assembly language routine shown in Exhibit 1 is self explanatory.

OS/360 ON-LINE TEST EXECUTIVE PROGRAM (OLTEP)

The On-Line Test Executive Program (OLTEP) together with associated on-line unit tests allows testing of System/360 I/O devices while operating under control of the Operating System/360 Supervisor. The program, when included in the operating system, will facilitate increased system availability and reduce the required maintenance time.

1. Description: Use of the OLTEP does not require the customer to relinquish the entire system to the Customer Engineer each time he desires to test a device. OLTEP is a system of testing that provides minimum interference with normal operations. The OLTEP with an associated I/O unit test are viewed by the OS/360 Supervisor as a problem program. Time independent tests of I/O units will be executed in the problem state and in a multiprogramming environment. The integrity of customer data is provided for with all tests. These unit tests directed by the OLTEP have the following purposes and usage:
   a) Periodic Check - to ascertain condition of an I/O unit
   b) Adjustment - to provide a tool to assist in making certain adjustments
   c) On Known Failures - to exercise the failing equipment to aid in further defining the malfunction
   d) After Repair - to verify the repair

   The OLTEP directs the execution of the on-line unit tests and acts as an interface between these tests and the Operating System/360 Supervisor.

2. Features: The following sections showing basic purpose are included in OLTEP.

a) CE Communications	-communicates with the operator/CE to define the job and provides for initialization and termination of testing.
b) Data Protection	-provides manual and program checks to insure the integrity of customer data.
c) Scheduler	-schedules and invokes the device tests that have been called for via CE communications.
d) On-Line Test Access Method	-provides for the required set up of control blocks, etc., before executing 'EXCP' - also interfaces with the OS with execution of the 'WAIT' macro.
e) DEB-DCB Subroutine	-sets up appropriate DEB and DCB for the device test.
f) DPRINT Subroutine	-provides for printing of unit test output messages via normal access method.
g) Compare Subroutine	-provides for checking of results from an I/O operation.

OS/360 PL/I DEBUGGING AID

```
*-------------------------------------------------------------------*
********                                                     ********
******                                                         ******
****                                                             ****
*                                                                   *
*-------------------------------------------------------------------*
*           THIS ROUTINE ACTS AS A CONDITION BUILT IN               *
*                                                                   *
*           FUNCTION I.E. IT CAN BE CALLED FROM AN 'ON-UNIT'        *
*                                                                   *
*           IF IT IS CALLED FROM ELSEWHERE IT WILL RETURN           *
*                                                                   *
*           THE VALUE 0(ZERO).                                      *
*                                                                   *
*           A LEGAL INVOCATION WILL RETURN, AS A FIXED              *
*                                                                   *
*           BINARY VALUE, THE STATEMENT NUMBER OF THE               *
*                                                                   *
*           STATEMENT CAUSING THE CONDITION TO BE RAISED            *
*                                                                   *
*                                                                   *
*           ONSTAT MUST BE DECLARED AS FOLLOWS -
*                                                                   *
*     DECLARE ONSTMT EXTERNAL ENTRY RETURNS(FIXED BINARY)           *
*                                                                   *
*                                                                   *
*          IT IS RECOMMENDED THAT ONSTMT RESIDE AS A MEMBER         *
*                                                                   *
*           OF  SYS1.PL1LIB.                                        *
*-------------------------------------------------------------------*
*-------------------------------------------------------------------*
****                                                             ****
******                                                         ******
********                                                     ********
ONSTMT   CSECT
         USING *,15             SET UP BASE
         STM   14,12,12(13)     SAVE REGISTERS
         LR    7,13             SAVE POINTER TO ON UNIT SAVE AREA
         LA    13,SAV           SET UP PTR TO OUR SAVE AREA
         ST    7,4(13)          CHAIN BACK
         LR    2,7              SET UP FOR LOOP
         L     3,0(1)           PICK UP ADDR OF TARGET FIELD
         L     2,4(2)           SKIP OVER ON UNIT DSA
         CLI   0(2),X'25'       IS IT SECONDARY LWS
         BC    6,OUTO           IF NO, EXIT
LOOP     L     2,4(2)           GO UP THE SAVE AREA CHAIN
         CLC   0(2,2),SAV+2     IS IT AN EXTERNAL SAVE AREA
         BC    8,OUTO           IF YES EXIT
         TM    0(2),X'80'       IS IT A DSA
         BC    8,LOOP           IF NO KEEP LOOKING
         TM    0(2),X'40'       ARE STMT NUMBERS PRESENT
         BC    8,OUTO           IF NO,EXIT
         MVC   0(4,3),96(2)     PICK UP STMT NUMBER
GO       LR    13,7             TIDY UP
         LM    14,12,12(13)     RESTORE REGS
OUT      BR    14               EXIT TO ON UNIT
OUTO     XC    0(4,3),0(3)      RETURN A ZERO VALUE
         BC    15,GO
         DS    0F
SAV      DC    X'900000FF'
         DS    17F
         END
```

Exhibit 1

h) Convert Subroutine -provides for conversion of HEX to EBCDIC and vice versa.

i) Random Data Subroutine -provides for generation of random data that may be used by unit tests.

j) CECOM -provides for operator/CE communication originating from the unit test.

3. Functions: To have the facility of on-line testing, the OLTEP and an associated SVC must be incorporated as an integral part of the Operating System. This may be accomplished either at the time of System Generation or via a system update. The I/O unit tests themselves comprise a third essential element of the on-line test facility. These unit tests will be released as a sequential data set on magnetic tape. Prior to use in the system, therefore, they must be translated to load module format in a partitioned data set. The OLTEP, residing on the LINK library, is called into memory via normal JCL input. The program, when loaded, will make requests via the console device to define the job to be executed. The information requested will include future communication intervals, devices to be tested, tests to be executed, and options to be exercised. When the job has been thus defined, a LINK will be executed within the OLTEP scheduler to load and begin execution of the initial test section. The execution of I/O unit tests will normally continue until all desired tests have been run on a device. If other units have been called for in defining the job, the test sections will be run again in the next unit, etc., until all the desired devices have been tested.

4. Data Protection: Data protection, within OLTEP, is devoted to insuring the integrity of customer's data when On-Line Tests are executed. Verifying that proper volumes have been mounted, setting up of extents, and establishing of file mask - mode set byte are all functions of data protection.

This protection is achieved by both manual and software checks. The manual checks, consisting of message output requiring Operator/CE response, will occur prior to execution of a test. These messages will relate to devices having demountable volumes, and will request a response indicating that the proper volume for testing has been mounted. Software checks, consisting of programmed tests to verify the volume are also made whenever possible. For certain device types, each test section, prior to execution, will be checked for the type of test to be performed. Based on the test type, the data protection section may allow only certain operations to be performed by setting the file mask - mode set byte, and may confine the test space to a specific area by establishing the extents.

Changing of file mask, mode set byte, extents--functions that are not normally performed by problem programs--are accomplished within OLTEP by use of a special SVC. A program check is made each time this SVC is executed to assure that it has been issued within the OLTEP. The user, therefore, may feel secure that these same functions cannot be initiated by normal problem programs when the on-line test facility is included in his system.

5. Equipment Configuration: The minimum system configuration required to run the OLTEP will be the same as that required to support the user's Operating System plus the additional unit or units to be tested.

Unit tests for System I/O devices will be released within the diagnostic area.

6. Performance: Approximately 1400 bytes of core will be required for the OLTEP with an additional 4000 bytes for the associated unit tests to be run. These unit tests are executed one at a time for a given device and therefore, 18K is the maximum main storage requirement. Approximately 350K bytes of external storage will be required on a direct access device used to store the partitioned data set comprising all I/O unit tests.

The execution time for a unit test will depend upon the type of operating system being used and the particular environment in which the test is being performed. The small amount of time sacrificed in testing, however, will be negligible with respect to the increased availability achieved with this type of testing.

7. Description of On-Line 2400 Tape Tests

a) Section T2400A (OS)

1. WTM at load point.
2. Series of REW's check load point bit on.
3. RDF TM - set unit exception.
4. RDB TM - set unit exception.

b) Section T2400B (OS)

1. WRT - RDB - RDF zeroes.
2. WRT - RDB - RDF ones.
3. BSR for correct positioning.
4. BSR over TM for unit exception.
5. BSR into load point for unit check.

c) Section T2400C (OS)

1. BSF over TM for positioning and status.
2. BSF over record into load point for unit check.
3. FSR for positioning and status.
4. FSR over TM for unit exception.

d) Section T2400D (OS)

1. FSF over 3 records and TM for positioning and status.
2. BSF at load point for unit check.
3. BSR at load point for unit check.
4. RDB at load point for unit check.
5. Multiple positioning commands for positioning.

e) Section T2400E (OS)

1. RDF record with lst character similar to TM for no unit exception.
2. RDB record with last character similar to TM for no unit exception.
3. RDB record with LRC similar to TM for no unit exception.
4. RDB record with CRC similar to TM for no unit exception.
5. RDB record with CRC and LRC similar to TM for no unit exception.

f) Section T2400F (OS)

1. WRT - RDB - RDF 256 characters (9 trk) - 64 characters (7 trk).
2. BSR over long record for positioning.
3. RDF with larger count than record for 1Ll, residual count.
4. RDB with short word count for 1Ll.

g) Section T2400G (OS)

1. SlLl flag ' on' to suppress 1Ll on large count.
2. 1Ll to break CCW chain.
3. SKIP flag 'on' for no data transfer.
4. Termination under count control for 1Ll.

h) Section T2400H (OS)

1. WRT - RDF - RDB random data.

i) Section T2400I (OS)

1. WRT 800 BPI.
2. WRT 556 BPI.
3. WRT 200 BPI.
4. WRT - RD at 800 BPI, DC and TRN off, odd parity.
5. WRT - RD at 800 BPI, DC and TRN off, even parity.
6. WRT odd parity, RD even parity.
7. WRT even parity, RD odd parity.

j) Section T2400J (OS)

1. WRT - RD random data

k) Section T2400K (OS)

1. Output TRN - WRT TRN 'on', RD TRN 'off'.
2. Input TRN - WRT TRN 'off', RD TRN 'on'.
3. WRT 256 characters TRN 'on', RD TRN 'off'.

l) Section T2400L (OS)

1. Translate 64 BCD characters to 64 EBCDIC characters. WRT TRN 'of', RDB TRN 'on'.
2. WRT 8-bit blank TRN 'on', RDB 6-bit substitute

blank TRN 'off'.

3. WRT 6-bit substitute blank TRN 'off', RDB 8-bit blank TRN 'on'.

m) Section T2400M (OS)

1. Output converter. WRT 18 bytes DC 'on', RD DC 'off'.
2. Input converter. WRT 32 bytes DC 'off', RD DC 'on'.
3. WRT-RD 256 characters DC 'on'.

n) Section T2400N (OS)

1. DC to write 1 byte as 2 tape characters.
2. DC to write 2 bytes as 3 tape characters.
3. DC to convert 1 tape character to 1 data byte and set unit check and DC check.
4. DC to convert 2 tape characters to 2 data bytes and and set unit check and DC check.

o) Section T24ØØO (OS)

1. DC to convert 3 tape characters to 3 data bytes and set unit check and DC check.
2. DC to convert 2 tape characters to 1 data byte, no unit check, no DC check.
3. DC to convert 3 tape characters to 2 data bytes, no unit check, no DC check.

NOTE: Sections 2400A to 2400G are 7 and 9 track tests.
Section 2400H is 9 track only.
Sections 2400I to 24ØØO are 7 track only.

8. 2311 On-Line Tests

a) Section T2311A (OS)
Tests basic communication between the 2311 + 2841 SENSE, READ, SEEK, RECALIBRATE. Commands are used.

b) Section T2311B (OS)
Tests difference counter and CAR bits both setting and being cleared.

c) Section T2311G (OS)
Tests the ability of the 2311 to seek to all cylinders and also perform a "worst case" seek test.

d) Section T2311D (OS)
Tests random seeks and writing using various data patterns.

e) Section T2311E (OS)
Tests multitrack operation, end of cylinder and rotation speed. Also provides two optional routines to seek between any two addresses or write home addresses on any track.

f) Section T2311F (OS)
Tests for correct ending conditions using various invalid seek or set file mask commands and and masks.

g) Section T2311G (OS)
Checks read count, space count and erase operations.

h) Section T2311H (OS)
Checks Search HA, write and read R0, write and read CKD, KD, and data.

DOS/360 SYSTEM GENERATION HINT

The following technique is a field contribution. Users should be aware that it has not been submitted to any formal IBM test and are expected to evaluate its usefulness in their own environment.

When running the job to catalog the user tailored supervisor, along with the critical programs such as Job Control, etc., do not include a / & card in the job stream. Place a pause card in its place in the job stream.

Although cataloging of the new supervisor takes place during the link edit run, the old supervisor is still in control of the system until the / & card has been read. By not having the / & card in the job stream, the user is assured that control will not pass to the new supervisor until he wishes it. Should any error in eith-

er linkage editing or any unrecoverable hardware situation occur, there would be no / & card to cause control to pass from the old supervisor to the newly cataloged supervisor. Should any error occur, do the following:

1. DO NOT RESET OR REIPL THE SYSTEM.

2. Correct the error condition. If a new assembly of the supervisor is required, it may be done at this time by submitting the deck for assembly starting with a // Job card. The new deck may be cataloged after the assembly by starting with a // Job card.

3. The system will once again come to the pause card. If no errors have occurred, submit the / & card to the reader. The following message will now appear on Syslog: 'New supervisor cataloged, reipl to continue'.

If no errors had occurred, when the system read the first // pause card, step 3 would have been performed immediately.

DOS/360 JOB TO JOB COMMUNICATION TECHNIQUE

The following is a field contribution which has not been submitted to any formal IBM test. Potential users are expected to evaluate its usefulness in their own environment and should be aware that no IBM maintenance will be available. Users should consider the implications involved in making changes to their system.

Since the communication region is cleared after each job, information cannot be passed from job to job. One user desired load up (daily) constants at the start of the day to be used by all the programs. He could have put a constant card in each deck to do this. However, a simple modification to the supervisor and the use of two macros provided the desired results. The FOPT macro is punched out and the modifications in Exhibit 1 are applied as stated in Columns 40-71. Then FOPT is recataloged and a new supervisor generated. The new supervisor will accept supervisory calls 218 and 219. These SVC's are used to get information into and out of the supervisor area. Exhibit 3 shows the macro SUPAREA which, when executed, loads the address of the 40 byte area into Register 1. The program can then move information out of the area. Since putting information into the supervisor area cannot be performed under storage protect by the problem program, SVC 219 was used to load the area. By using the STORESUP macro, shown in Exhibit 3, the modified supervisor will take the address in Register 1 and move 40 bytes into the supervisor, then return to the user program.

The macros and SVC's use Registers 9 and 10 because the locations of the save area for these registers after issuing a SVC are easy to locate. The macros use 9 and 10, but restore them to the user. The problem program can use 9 and 10, in any way it wishes, even as base registers without restriction. If a larger area is needed, the only change is to the length of the DC names KKTEMP in Exhibit 1. These macros and the supervisor area have been used for over three months, on a system with storage protect, without problems.

DOS/360 JOB TO JOB COMMUNICATION TECHNIQUE

IBM

IBM System/360 Assembler Coding Form

PROGRAM | PROGRAMMER | DATE | PUNCHING INSTRUCTIONS | GRAPHIC PUNCH | PAGE | CARD ELECTR

STATEMENT

Name	Operation	Operand	Comments
SVC219	ST	9,KKREG	THIS GROUP OF INSTRUCTIONS ARE
	L	9,ERA	TO BE PLACED FOLLOWING INST.
	MVC	KKTEMP,Ø(9)	'B ERRGO' WHICH IS ABOUT INST NO
	L	9,KKREG	MS14Ø189 IN THE PRINTOUT OF THE
	BR	R9	FOPT MACRO.
SVC218	MVC	ERA(4),KKCONST	
	BR	R9	
KKTEMP	DC	CL4Ø' '	
KKCONST	DC	A(KKTEMP)	
KKREG	DS	F	
	B	ERR22EX	THIS GROUP OF INST. REPLACE
ERR21	CLI	SVOLDP+3,218	'ERR21 BCTR RB,Ø ILLEGAL SVC' NO.
	BE	SVC218	ABOUT MS14Ø2Ø2 IN THE PRINTOUT
	CLI	SVOLDP+3,219	OF THE FOPT MACRO.
	BE	SVC219	
ERR22EX	BCTR	RB,Ø ILLEGAL SVC	

Exhibit 1

IBM IBM System/360 Assembler Coding Form

PROGRAM | PUNCHING INSTRUCTIONS | GRAPHIC PUNCH | PAGE CARD ELECTR

PROGRAMMER | DATE

STATEMENT

Name (1–8)	Operation (10–14)	Operand (16–71)	Comments (50–71)
	CATALS	A.SUPAREA	THIS SET OF CARDS CATALOGS
	BKEND	A.SUPAREA	THE SUPAREA MACRO IN LIB.
	MACRO		
&NAME	SUPAREA		
&NAME	STM	9,10,QQQ1&SYSNDX	THIS MACRO WILL GET THE
	B	QQQ2&SYSNDX	ADDRESS OF THE SUPERVISOR
QQQ1&SYSNDX	DS	2F	AREA AND PUT IT IN REG 1.
QQQ2&SYSNDX	LA	10,QQQ1&SYSNDX	REG'S 9&10 ARE USED BUT
	SVC	218	THEY ARE STORED AN RESTORE
	LR	1,9	WITH OUT ANY EFFECT ON THE
	LM	9,10,0(10)	REST OF THE PROGRAM.
	MEND		
	BKEND		

Exhibit 2

DOS/360 JOB TO JOB COMMUNICATION TECHNIQUE

IBM

IBM System/360 Assembler Coding Form

PROGRAM | PUNCHING INSTRUCTIONS | GRAPHIC | PAGE

PROGRAMMER | DATE | PUNCH | CARD ELECTRC

STATEMENT

Name	Operation	Operand	Comments
	CATALS	A.STORESUP	THIS SET OF CARDS CATALOGS
	BKEND	A.STORESUP	THE STORESUP MACRO IN LIB.
	MACRO		
&NAME	STORESUP		
&NAME	STM	9,10,QQQ1&SYSNDX	THIS MACRO WILL GET THE
	B	QQQ2&SYSNDX	DATA AT THE ADDRESS IN REG
QQQ1&SYSNDX	DS	2F	1 AND MOVE IT INTO THE
QQQ2&SYSNDX	LA	10,QQQ1&SYSNDX	STORAGE AREA IN THE
	LR	9,1	SUPERVISOR. REG'S 9&10 ARE
	SVC	219	USED BUT RESTORED
	LM	9,10,0(10)	
	MEND		
	BKEND		

Exhibit 3

BOS/360 RPG CHAINING RECORD NOT FOUND ROUTINE

The following is a field contribution which has not been submitted to any formal IBM test. Users are expected to evaluate its usefulness in their own environment.

Under BOS RPG using chaining, when a record is not found in the chained file, the message 5885A is typed. The operator can terminate by typing a 0 or 1, bypass the chaining record by typing a 2 (program branches to GETREC), or he can bypass the chained record by typing a 5 (program branches to DETAIL CALCS).

One user wanted operator intervention eliminated; he always wants to go to DETAIL CALCS where he will check indicators to determine the status. (Model 20 Disk RPG works this way).

The attached assembler (V1-M8) subroutine was written and cataloged into the relocatable library. An exit is taken to it once in each program that does chaining, usually as the first detail calculation conditioned, for example, on date card resulting indicator.

The routine (See Exhibit 1) finds the one, two, or three places in core where RPG has generated the SVC that generates the message and puts 16 bytes of NOP's in its place. The cross reference listing is shown in Exhibit 2.

Note that f'80' is added to GR3 before comparing starts so that LIT will not be compared to itself. Other users might have to increase this if the program contains more than one subroutine and NO5885 is not at program load point. F'16000' must be higher than the probable location of the SVC for the last chained file, but less than core capacity.

```
LOCATN  OBJECT CODE    ADDR1 ADDR2 STMNT   SOURCE STATEMENT   360B AS 309  V1M8

                                    0001          ADOPTN CROSSREF
                                    0002 NO5885   AFILE LIBRARY,RETAIN
                                    0003 NO5885   START
000000 05F0                         0004 BALR     BALR  15,0
000002                              0005          USING *,15
000002 5030 F032       00034        0006          ST    3,F
000006 5A30 F03A       0003C        0007          A     3,FIG
00000A D505 F042 3000  00044 00000  0008 CLC      CLC   LIT,0(3)
000010 4780 F022       00024        0009          BE    HIT
000014 5A30 F036       00038        0010 UP       A     3,ONE
000018 5930 F03E       00040        0011          C     3,TOP
00001C 4780 F02C       0002E        0012          BNL   DONE
000020 47F0 F008       0000A        0013          B     CLC
000024 D20F 3000 F048  00000 0004A  0014 HIT      MVC   0(16,3),NEW
00002A 47F0 F012       00014        0015          B     UP
00002E 5830 F032       00034        0016 DONE     L     3,F
000032 07FE                         0017          BR    14
000034 00000000                     0018 F        DC    F'0'
000038 00000001                     0019 ONE      DC    F'1'
00003C 00000050                     0020 FIG      DC    F'80'
000040 00003E80                     0021 TOP      DC    F'16000'
000044 0A02F5F8F8F5                 0022 LIT      DC    X'0A02F5F8F8F5'
00004A 4700F0024700F002             0023 NEW      DC    X'4700F0024700F002'
000052 4700F0024700F002             0024          DC    X'4700F0024700F002'
000000                              0025          END   BALR
```

Exhibit 1

BOS/360 RPG CHAINING RECORD NOT FOUND ROUTINE

CROSS REFERENCE LISTING - DISK ASSEMBLER,8K

SYMBOL	LEN	VALUE	DEF	CROSS-REFERENCE	
BALR	002	000000	0004	0025	
CLC	006	00000A	0008	0013	
DONE	004	00002E	0016	0012	
EIG	004	00003C	0020	0007	
F	004	000034	0018	0006	0016
HIT	006	000024	0014	0009	
LIT	006	000044	0022	0008	
NEW	008	00004A	0023	0014	
NO5885	000	000000	0003		
ONE	004	000038	0019	0010	
TOP	004	000040	0021	0011	
UP	004	000014	0010	0015	

Exhibit 2

S/360-USING THE IBM EXECUTARY TO IMPROVE OPERATIONS

Several customers, particulary in the larger systems and multiprogramming installations, have placed IBM Executaries at the console of their systems. This provides a quick and effective method for recording positively the conditions existing at the time of occurrence of any incident which affects processing.

The IBM Executary is used by customer operators, operator supervisors, and IBM Customer Engineers and System Engineers. Data recorded includes the time of day, system status, job starts, completions and stops, unscheduled interrupts, actions taken, time lost, and other information which seems pertinent at the time. Each moring the recording for the previous twenty-four hours, is typed, reproduced, and distributed. Then, each day a meeting is held by customer and IBM representatives to discuss each problem, its cause and solution. Any problem which is not resolved is identified and responsibility for follow-up is given to a member of the group.

The procedure is very helpful in providing a history of events; it provides a document for improving communications between customer and IBM personnel concerned with identifying and solving hardware, software, operator, programming and related problems. The history in this concise form allows identification of recurring incidents and also provides a method for insuring that problems have been resolved.

The result is better understanding and improved production and turn-a-round time.

S/360 USING TRANSLATE INSTRUCTION TO REARRANGE DATA

The following technique is a field contribution which has not been submitted to any formal IBM test. Potential users are expected to evaluate its usefulness in their own environment. The Translate instruction is normally used to convert from one character set to another. An example would be from BCD to EDCDIC. In the example below, Translate is used to alter the position of characters in a data record. The data is unchanged. Specifically, the example deals with reversing the sequence of characters in a ten-character record--the byte in the tenth position is moved to the first position of the target field, the ninth to the second, eighth to the third, etc.

```
GET     input file, INPUT

MVC     TARGET, X'09080706050403020-
        100'
TR      TARGET, INPUT
TARGET  DC      CL10
INPUT   DC      CL10
```

The GET is included here to emphasize that the input data is variable; any characters in the EBCDIC set will do. The MVC sets up the TARGET field for the Translate instruction. When the TR is executed, the byte at INPUT +09 replaces the 09, the byte at INPUT +08 replaces the 08, and so forth.

This approach is, of course, not limited to reversing data fields. Any logical rearrangement is possible. For example, COS uses this approach to extract every other byte of a card input area when handling Column Binary data. (S/360 reads column binary cards into one continuous 160 byte area. The 1401 reads such data into two discontinuous areas at locations 401 and 501. COS "Translates" 360 positions 1, 3, 5,... to 1400 positions 401, 402, 403....)

The TRANSFORM verb in COBOL uses the Translate instruction and a similar effect can be coded in COBOL. The following are the COBOL statements to reverse the position of data in a 10 byte input record.

```
DATA DIVISION.
        INPUTA          PICTURE    X(10).
        TARGET          PICTURE    X(10).
PROCEDURE DIVISION.
        MOVE 'JIHGFEDCBA' TO TARGET.

        TRANSFORM TARGET FROM 'ABC-
        DEFGHIJ' TO INPUTA.
```

To paraphrase the COBOL Language Specifications manual: "If equal in length, any character in TARGET equal to any character in the non-numeric literal is replaced by the character in the corresponding position of INPUTA."

It should be pointed out that this statement necessarily generates a significant amount of code. It will require about 10 instruction executions to move a character from INPUTA to TARGET. If coding speed is more important than execution time, as in a one time conversion program, this can be a very useful approach.

NEW TYPE III PROGRAMS *

The following are the abstracts of programs which have been recently made available from the Type III library.

The program, along with its complete abstract, will be incorporated in subsequent issues of the Catalog of Programs.

Programs may be obtained by submitting a properly completed "General Program Request Card" (Form Number 120-1145-1) to the Program Information Department, 40 Saw Mill River Road, Hawthorne, New York, 10532.

These programs and their related documentation are distributed by IBM in the author's original form and have not been subjected to any formal testing.

Any discussion of Type III Programs must emphasize the following points:

1. **Type III programs are not part of the IBM product line, as are Programming Systems (Type I) or Application Programs (Type II).**
2. **Type III programs have not been subjected to any formal product test.**
3. **Recipients of Type III programs are expected to make the final evaluation as to the usefulness of the programs in their own environment.**
4. **There is no committed maintenance for Type III programs. However, any changes the author chooses to make will be announced in subsequent issues of the Catalog of Programs.**

* NOTE: THE CUSTOMER MUST BE INFORMED THAT THE ABOVE APPLIES TO ANY OF THE FOLLOWING TYPE III ABSTRACTS FURNISHED TO HIM.

COS/40-COMPATIBILITY OPERATING SYSTEM. Additional information is provided below since the announcement write-up in Installation Newsletter 67-26 for Model COS/40 did not include the minimum machine requirements. These are as follows:

1) 1400 compatibility feature--#4457

2) Disk compatibility--#9710

3) Decimal arithmetic--#3237

4) For a generated DOS supervisor size greater than 8K-- RPQ E54022 for core relocation to 16K.

5) First selector channel for load mode magnetic tape operations.

S/360 DOS/TOS UPDATE PROGRAM TO MAINTAIN MULTIPLE PROGRAM MODULES ON MAGNETIC TAPE. This program provides the facility to keep many source program modules or other data such as job control procedures on a single tape. Each module on the tape is separated from the others and has its own sequence numbers starting at 000000. In the past, most update programs have been used to maintain files of deblocked card images which were used directly as input to the assembler or other language translator. The update program uses blocked old and new master tapes (40 cards/block) and produces a third deblocked tape to be used by the language translators. A most important feature is that only those modules specified by the user are written on the deblocked tape. A 32K System/360 Model 30 or larger system having 3 tape drives for DOS or 4 for TOS is required. The program is written in Assembler language.

Ordering Procedure: Order File Number 360D-00.3.012.

The program material can be obtained on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

The requester must indicate whether a 9 track or 7 track DTR is required. If not specified, a 9 track will be provided.

No tape submittal is required. The DTR will be provided by the Library.

S/360 DISK PACK AUTOMATIC CATALOGING. DPAC is a three part programming system for automatically keeping a current catalog of disk space utilization. Minimum configuration is a 2030-D with the decimal feature under BOS. It can be modified to run under DOS.

Phase 1 formats the catalog on the system resident 1316 and enters a label in the system resident volume table of contents, so that the catalog can be accessed logically.

Phase 2 catalogs on the system resident pack information from all on-line 1316/s when any file is created.

Phase 3 prints on request a two page report per 1316/s. Page 1 shows the file label, volume sequence number of the originating volume, volume number, creation and expiration date, extent type and sequence, and the upper and lower limits of all the files on the 1316. If a file has expired, it is flagged with an asterisk. Page 2 is a listing of the available cylinders on the 1316.

Odering Procedure: Order File Number 360D-00. 4. 010.

The program material can be obtained in card form or on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

The requester must indicate whether a 9 track or 7 track DTR is required. If not specified, a 9 track DTR will be provided.

No tape submittal is required. The DTR will be provided by the Library.

S/360 PROCESS MODELING AND CONTROL-DESIGN PACKAGE. This is a program package, comprising two FORTRAN programs to analyze continuous industrial processes. The first program analyzes a record of repeated measurements, equally spaced in time, of a number of manipulated variables and one dependent variable. It produces a representation of the sample in the form of mathematical and statistical models of the process and its disturbances. The second program calculates various conventional process and disturbance characteristics, and/or an optimal control law. Application is limited to processes that can be described by a linear, single-output, time-invariant, dynamic model, superposed by a normal random time-series with time-invariant power spectrum. Diagnostic tests are included.

The package should reduce systems engineering work needed to apply process control computers, in particular when the process has complicated dynamics and significant time-delays, is slow-reacting, and substantially disturbed during running conditions. Sufficient machine configuration is S/360-40 with DOS or TOS. Core storage requirement is 128 k bytes. Source language is Basic FORTRAN IV.

ORDERING Procedure: Order File Number 360D-13. 4. 004.

The program material can be obtained on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

The requester must indicate whether a 9 track or 7 track DTR is required. If not specified, a 9 track DTR will be provided.

No tape submittal is required. The DTR will be provided by the Library.

S/360 NONLINEAR PARAMETER ESTIMATION AND PROGRAMMING. The program is designed to solve the following problems: 1) Estimate unknown parameters in nonlinear mathematical models, using any of the following techniques: least squares, weighted least squares, maximum likelihood, bayesian estimation. Special provisions are included for models involving solutions of ordinary differential equations, and for chemical reaction kinetics equations. Constraints may be imposed on the parameter values. 2) Solve nonlinear programming problems. 3) Solve simultaneous equations, two point boundary value problems, and other problems which can be cast in the form of one of the two above mentioned forms.

The program is written in the lowest level FORTRAN IV language. It can be run on the IBM S/360 under the OS or BPS monitors, on the IBM 7090 or 7094 computer under IBSYS, etc.

Ordering Procedures: Order File Number 360D-13. 6. 003.

The program material can be obtained on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

The requester must indicate whether a 9 track or 7 track DTR is required. If not specified, a 9 track DTR will be provided.

No tape submittal is required - the DTR will be provided by the Library.

S/360 ADVANCED LIFE INFORMATION SYSTEM MASTER RECORD UTILITY PROGRAM. This program is intended as a conversion aid for the Advanced Life Information System. Any field in the ALIS master record (as it exists on tape) may be changed by use of this program with the IBM Disk Operating System. The user specifies the following information in a transaction card for each field to be changed: policy number, trailer ID, trailer number (if more than one trailer exists on the master record with the same ID), field position within the trailer and field contents before and after the change. The "before" field in the transaction must compare equal to the old master field before a change will take place. A register is produced which shows field contents both before and after a change, as well as an error code for those changes which could not be made. A "Change ALL" option is provided, by which a field may be changed in every master record in the file.

Machine Configuration : minimum configur- for the Disk Operating System, plus two tape drives.

Source Language: S/360 Assembly Language.

Ordering Procedure: Order File Number 360D-04.3.002.

The program material can be obtained in card form or on one 9 track or 7 track (Data Conversion feature requied) Distribution Tape Reel (DTR).

The requester must indicate whether a 9 track or 7 track DTR is required. If not specified, a 9 track DTR will be provided.

No tape submittal is required. The DTR will be provided by the Library.

S/360 COBOL CROSS REFERENCES AND RESEQUENCING PROGRAM. This is a set of two IBM S/360 Programs that will provide a cross reference listing, by sequence number, of COBOL source programs. Data and paragraph names are identified in alphabetic sequence and all references to these names by procedure division statements are noted. All literals (except 0 and 1) and selected COBOL words are listed indicating their location within the procedure division. The user must provide a sort program of his choosing. In addition to providing a corss reference, provisions are made to:

1. Resequence a COBOL source deck.
2. List a COBOL source deck.
3. Re-identify a COBOL source deck.
4. Any combination of the above - indicating cross reference.
5. Reproduce and/or list any card deck.
   Multiple COBOL source programs can be be processed in a 'batch' environment.

The programs are written in COBOL under control of DOS. However, they could be adapted to any operating system by the user re-compiling and creating the necessary job stream and associated control cards.

The corss reference listing can be extremely helpful in debugging COBOL programs. The resequencing and/or re-identifying functions should also prove useful as additional entries are made tothe source program during the debugging stage.

Minimum system requirements are:

- 2540 Card-Read-Punch
- 1403 Printer
- At least 1 tape drive
- Any other components required for the sort the user chooses. This will also determine core requirements.

NOTE:

By changing the 'select' statements in the COBOL source decks and compiling, any type of Card-reader, Punch and Printer could be used. A punch is required only if the input card decks are to be reproduced.

Ordering Procedure: Order File Number 360D-03.6.006.

The program material can be obtained in card form or on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

The requester must indicate whether a 9 track or 7 track DTR is required. If not specified, a 9 track DTR will be provided.

No tape submittal is required. The DTR will be provided by the Library.

CARD TO PAPER TAPE/MAGNETIC TAPE AND MAGNETIC TAPE TO PAPER TAPE PROGRAMS UNDER OS/360. The card to paper tape/magnetic tape program will punch paper tape or build a magnetic tape from Hollerith or column binary card input. The magnetic tape to paper tape program will punch paper tape from magnetic tape input. Machine requirements: IBM System/360 under the Operating System; Decimal Arithmetic; IBM 1012 Paper Tape Punch; IBM 2821 Control Unit with column binary feature; IBM 1403 Printer; IBM 2400 Tape Unit with 9-track read/write head; IBM 2540 Card Reader/Punch.

Ordering Procedure: Order File Number 360D-01.1.001.

The program material can be obtained on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

The requester must indicate whether a 9 track or 7 track DTR is required. If not specified, a 9 track DTR will be provided.

No tape submittal is required. The DTR will be provided by the Library.

A SYSTEM/360 JOB ACCOUNTING AND SCHEDULING AIDS PACKAGE TO RECORD, EVALUATE, AND CONTROL TIME (REACT). This package is a collection of programs that Record, Evaluate, and Control time. Three programmed steps account for computer time, calculate utilization, analyze operations by shift and provide data on job performance for scheduling purposes. The Daily Usage Log similarly illustrated in the S/360-20 User's Guide is the source document for all the programs. Date, Job ID, department, use type, operator, clock time, meter time, input volume, and sequence performed are recorded. The first step edits the data for accuracy. The second calculates utilization according to job classification such as production, reruns, testing, idle time, etc., within the user's own shift periods. A third program step times completed jobs giving average job time per unit of input volume. A computer scheduling form is included. These programs are specifically designed to aid in the effective use of available computer time. The basic program material is in RPG source form. It can be run under any S/360 Programming System supporting RPG. I/O requirements are one card reader, one card punch, and one printer (100 Print positions). Minimum core requirements are 8K for Card Model 20, 12K for Tape or Disk Model 20, 16K for BOS-TOS-DOS.

Ordering Procedure: Order File Number 360D-15.0.003.

The program material can be obtained in card form or on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

The requester must indicate whether a 9 track or 7 track DTR is required. If not specified, a 9 track DTR will be provided. No tape submittal is required. The DTR will be provided by the Library.

S/360 NET EXPLOSION TECHNIQUE OF INVENTORY CONTROL FOR MANUFACTURES. The Net Explosion Technique of Inventory Control for Manufacturers (NETIM) is a system of programs to explode net inventory requirements using a Bill of Material File.

The gross requirements for products are first "netted" against available inventory, and the net requirements thus generated are then exploded into requirements for the component parts and assemblies using a Bill of Material file. The system includes programs for establishing and maintaining a card Bill of Material file for use in the explosion process.

All programs are written in RPG for a card IBM S/360 Model 20 system with 8,000 positions of core storage, printer, MFCM, and an auxiliary card reader.

Ordering Procedure: Order File Number 360D-23.3.002.

The program material can be obtained in card form or on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

The requester must indicate whether a 9 track or 7 track DTR is required. If not specified, a 9 track DTR will be provided.

No tape submittal is required. The DTR will be provided by the Library.

S/360 "OPTIM" - ORDER POINT TECHNIQUE OF INVENTORY FOR MANUFACTUERERS. This program package may be utilized to establish a basic order point inventory control system on the IBM S/360-20 Card System. It is designed to be easily understood, installed, and operated in order to provide a logical step toward better inventory control for IBM customers. Three "RPG" programs have been provided to be used "AS IS" or modified to meet individual requirements. The programs are an Inventory Update, a Reorder Analysis, and a Reaveraging (Usage) that uses the Exponential Smoothing method. Documentation includes "RPG" source list, object program decks, operating procedure, and a sample problem. Systems configuarion is 8K, 360-20, 2203 Printer, 2501 Card Reader, and a 2560 Multifunction card machine with an interpreting feature.

Ordering Procedure: Order File Number 360D-25.0.001.

The program material can be obtained in card form or on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

The requester must indicate whether a 9 track or 7 track DTR is required. If not specified, a 9 track DTR will be provided.

No tape submittal is required. The DTR will be provided by the Library.

S/360 MODEL 20 MULTI-FUNCTION UTILITY PROGRAM. This program was written to provide a convenient method for Model 20 users to perform the basic card functions. The program will: 1) 80-80 List; 2) 80-80 Reproduce; 3) Interpret Input Cards; 4) Interpret Output Cards; or any combination of these functions under control of data switch 1 on the Model 20 console. This program, with its ability to perform multiple functions on one pass, and its ease of operation, should make it very useful in all Model 20 installations:

Minimum configuration: 4K Model 20 and 2560 MFCM with printing ability.

Ordering Procedures: Order File Number 360D-00.1.012.

Distribution will be in card form only.

S/360 MODEL 20 DISK AND TAPE PROCESSING SYSTEM MACROS. These 26 macros are a modification of the BPS/BOS package 360D-03.7.003 and DOS/TOS package 360D-03.7.008. Macros available are ADDP, SUBP, MULTY, DIVID to provide automatic multiple parameter decimal aligning arithmetic. ZEROP to zero up to 49 fields, BLANK to blank up to 24 fields of any length, SHIFT packed fields left or right 1 to 31 digits with optional rounding, PAC multiple parameters and insure a valid sign, UNPAC multiple parameters, EDIT to generate a hex edit mask and perform the editing, COMP to compare packed or zoned fields and generate branches, RANGE to test the magnitude of a field for two limits, IFNUM to validate a field for pure numeric content, 7 byte-switch defining, setting, testing, and first-time assembly language coding for the usual RPG oriented mod 20 user, reduced coding errors, easier keypunching and verification, and faster debugging. Since the model 20 macros are a conversion of an existing package they should be error free. Macros are upward compatible to BPS/BOS/TOS/DOS on System/360 models 30 thru 75. MACHINE REQUIREMENTS: Designed to be cataloged into the Model 20 DPS and TPS, and will function on any Model 20 the DPS/TPS will function on. SOURCE LANGUAGE: Model 20 DPS/TPS macro language.

Ordering Procedure: Order File Number 360D-03.7.015.

The program material can be obtained on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

The requester must indicate whether a 9 track or 7 track DTR is required. If not specified, a 9 track DTR will be provided.

No tape submittal is required. The DTR will be provided by the Library.

S/360 LOADING MODEL 20 OBJECT DECKS WITH THE TAPE MONITOR. This program, once cataloged into the core-image library of the Tape Programming System for the Model 20, allows a user to execute Model 20 object programs in card form under control of the tape-resident monitor. This is an option not currently available in the Tape Programming System. The

The purpose is to allow execution of object decks in a job stream with other programs that may need the facilities of the tape-resident system, such as the compilers. The Program performs the same loading functions as the card-resident monitor of the Tape Programming System: TXT, END, XFR, and REP cards are processed and all others are bypassed.

Since the program resides between the monitor and the lowest possible starting address for RPG programs, it effectively takes no additional core storage. To use this program with BAL object decks, the BAL program must be specified with a start address of 2144 or greater.

Machine configuration: Same as required for the Tape Programming System. Three tapes are required to catalog the program into the program library of the system tape.

Source Language: Model 20 Basic Assembler Language.

Ordering Procedure: Order File Number 360D-01.1.002.

The program material can be obtained in card form or on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

The requester must indicate whether a 9 track or 7 track DTR is required. If not specified, a 9 track DTR will be provided.

No tape submittal is required. The DTR will be provided by the Library.

AN AUTOMATED VAN LOADING PROGRAM FOR THE IBM 1130. The Van Loading program is designed to match a system configuration to a file of the physical characteristics of each component and to then identify the location within the van which will provide the maximum transit protection in terms of each ride, padding, elimination of box shifting and accessible tie downs. This location will be determined by parameters, built into the program, which express the characteristics of each type of van. The program will be expanded to include loading guides for both railroad freight car and aircraft shipments.

Machine language will be 1130 FORTRAN for use on an IBM 1130 with 2K core storage and disk pack. Output will be punched cards, which, when listed indicate box loading sequence by alternating sides of the carrying space. Other output could be as cards defining vectors for use on a line drawing device.

Ordering Procedure: Order File Number 1130-23.1.003.

Distribution will be in card form only.

1130 FORTRAN - CALLABLE SUBROUTINES TO READ AND/OR PUNCH NUMERICAL CONTROL TAPES. A set of 1130 FORTRAN and assembler subroutine which allow the user to read and/or punch any character, one at a time, on paper tape. Subroutines are also included to break a floating point number into a series of P/T characters for subsequent punching.

These subroutines were developed to allow the reading and punching of P/T characters which FORTRAN couldn't normally handle in the numerical control area but can obviously be used in any paper tape application where non-IBM codes are used.

Configuration: 4K 1130, 1055, 1134.

Ordering Procedure: Order File Number 1130-23.4.003.

Distribution will be in card form only.

S/360 DOS/360 CIVIL ENGINEERING COORDINATE GEOMETRY (COGO). DOS/360 Civil Engineering Coordinate Geometry (COGO) is a direct conversion from COGO for the IBM 1130. The machine configuration required, in addition to the requirements for DOS FORTRAN, is a 32K 360 (with a 6K or 8K DOS Supervisor) and

2 cylinders on a 2311 disk. The program is written in FORTRAN AND BAL. See the IBM 1130 COGO User's Manual (H20-0301) for a description of COGO.

Ordering Procedure: Order File Number 360D-16.2.003.

The program material can be obtained on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

The requester must indicate whether a 9 track or 7 track DTR is required. If not specified, a 9 track DTR will be provided.

No tape submittal is required. The DTR will be provided by the Library.

1401 SIMULATOR FOR SYSTEM/360. This program allows the execution of 1401 programs on a System/360 Model 30, 40, 50, 65, 75 without using any special hardware or re-programming. 1401 features supported are advanced programming, sense switches, tapes, multiply, divide, -16K core, and all standard instructions except Select Stacker. Operator control is through the 1052 and tape and includes sense switch assignment, load from cards or tape, display and alter core, start the reset, start, clear core, set and clear I/O check switch, and write tape mark. The program is written in OS/360 Assembler language, but runs stand-alone. It should be especially useful to customers converting from a 1401 to a System/360, Model 50 or higher. The simulator requires Standard Instruction set, 65K core, one each 1052, 1403, 2540.

Ordering Procedure: Order File Number 360D-11.1.017.

The program material can be obtained on one 9 track or 7 track (Data Conversion feature required. If not specified, a 9 track DTR will be provided. No tape submittal is required. The DTR will be provided by the Library.

1401 REAL ESTATE TAX BILL SYSTEM FOR THIRD CLASS PENNSYLVANIA COUNTIES. To calculate and print real estate tax bills given the assessment and mileage rates. Tp provide summary cards during calculation time for later generation of monthly reports. Machine configuarion - 1401, 4K, 1403, 1402, advanced programming (advanced programming can be eliminated) source language - autocoder.

Ordering Procedure: Order File Number 1401-14.0.026.

Distribution will be in card form only.

1800 TSX MAINLINE TRACE PROGRAM. The 1800 TSX Mainline Trace Program consists of two subroutines which can be built with a mainline coreload to enable either full trace or checkstop trace of that mainline. The trace set-up subroutine (TRSUP) is called by a statement in the mainline, and communicates with the operator to set up the trace limits. The trace interrupt subroutine (TRACI) is then executed for every mainline level instruction. The program is written in 1800 Assembler language and is available in source form only.

Ordering Procedure: Order File Number 1800-04.2.001.

Distribution will be in card form only.

1800 TSX DISKN OPERATION COMPLETE MODIFICATION. This program provides a modification to the 1800 TSX system DISKN subroutine which will allow the user to specify a subroutine to be executed upon completion of the called disk operation. This modification consists of several 1800 Assembly language source cards which must be inserted in the source deck of the 1800 TSX TASK program.

Ordering Procedure: Order File Number 1800-03.4.003.

Distribution will be in card form only.

1800 TSX CIRCULATING TRACE TABLE SUBROUTINE. The Circulating Trace Table Subroutine (TRTSX) is used to create a dynamic trace record of programs being executed under the 1800 Time Sharing Executive system. TRTSX calls are assembled or compiled with process programs. Whenever one of these programs is then executed, TRTSX generates a four word trace record in the trace table in SKELETON COMMON. This record contains the current interrupt level being processed,

the system clock value, the address of the call, and the user data parameter. If the table is filled, the oldest record is replaced with the new trace record. Size of the table and location in common is optional. This table can be dumped at any time to provide a time dependent trace of programs executed.

Ordering Procedure: Order File Number 1800-04.2.002.

Distribution will be in card form only.

IBM 1800 CONSOLE INTERRUPT PROGRAM. The IBM 1800 Console Interrupt Program consists of a group of subroutines which enable operator controlled execution of several functions which are useful in debugging, modifying, and communicating with programs being executed under the process mode. These functions are 1) entry of hex, decimal, and floating point data, 2) dump of hex, decimal, and floating point data to console printer and list printer, 3) set up of full trace or check-stop trace, 4) setting of program switches in the COMMON area, 5) execution of any user program. All features are optional and can be selected or omitted through use of user coded EQUATE cards assembled with the CONSOLE INTERRUPT PROGRAM. The program is provided in source cards written in 1800 ASSEMBLY Language. The user must assemble the program and then build the coreloads or skeleton including the console interrupt program.

Ordering Procedure: Order File Number 1800-04.1.001.

Distribution will be in card form only.

1800 TSX TYPEN MODIFICATION FOR IN CORE MESSAGE BUFFERING. This program provides a modification to the 1800 TSX System (1800-OS-001) which will allow buffering of 1053 messages in core rather than on disk. This will eliminate the wait loops which can occur on the call level and the typewriter levels whenever message buffering to disk takes place. Options selectable by the user through use of EQUATE cards at assembly time are 1) amount of core allocated for buffering for each 1053, 2) priority messages, 3) action taken when buffer overflows. This program consists of a deck of 1800 Assembly language source statements which must be inserted in the 1800 TSX TASK program before assembly of TASK.

Ordering Procedure: Order File Number 1800-03.4.002.

Distribution will be in card form only.

IBM 1800/1896 PROCESS COMMUNICATION AND CONTROL SYSTEM. The 1800/1896 Process Communication and Control System is a teleprocessing support package for the 1800 System utilizing the 1896 Communications Adapter (RPQ C08413). The system works within TSX and provides support for up to sixteen lines with sixteen terminals per line. The system is designed to support 1070, 1050, 1030, and 2740 terminals, and 1800-360, 1800 - 1800 communication. It is currently fully implemented only for 1070 terminals. The author states implementation for other terminals and computer to computer communication will be reflected in the 1800 catalog of programs.

Features include printed output via FORTRAN WRITE statements, queued I/O, on-line diagnostics, all user interface via FORTRAN, and automatic code conversion. Minimum system requirements are 16K of core and TSX. Skeleton requirements vary from 2000 to 3000 words in addition to TSX requirements. Interrupt requirements are exclusive use of level zero and the program-settable interrupts of two lower levels. Up to sixteen typewriters are supported with FORTRAN WRITE statements. The system is currently operating successfully in both multi-terminal and multi-line systems with full TSX functions.

Ordering Procedure: Order File Number 1800-03.4.004.

To obtain program material, submit one 2400 foot reel of magnetic tape. Specify whether 7 or 9 track recording is required. If not specified, 9 track recording will be used.

The required tape may be ordered from IBM or supplied with your request for the program.

IBM Installation Newsletter

February 9, 1968 Issue No. 68-02

CONTENTS

(G means Green Section)

Distribution: Branch Offices-DP Management, Salesmen, Systems Engineers, FE Managers, Regions, Districts, Education Centers, Field Systems Centers, Federal Systems Center, FE Area Offices, DPD HQ, FED HQ, WTC.

*Requires Immediate Attention

For IBM Internal Use Only

DPD HQ, White Plains, N. Y.

SELECTED SHORT SUBJECTS

The purpose of Selected Short Subjects is to bring together and highlight concise, factual and timely information which will indicate that action is to be taken by the IBM representative whose accounts are affected.

1. IBM 1800, 1070, and 1080 Systems Assistance

The SDD C/DA (Control and Data Acquisition) resident support personnel to be the first point of contact for assistance with system design problems, details relative to previously quoted RPQ's and sales call assistance as outlined in DPD Blue Letter 266-67.

To supplement this service, when necessary, a telephone system designed to provide the Field with effective response to our customer's requests for Special Equipment and assistance with solutions to unique C/DA application has been established within SDD Special Systems Design (Department 590) San Jose.

Your request, realistically defined, will assure you of receiving assistance.

Phone: San Jose Tie-Line 132, or Area Code 408, 227-7100, Extension 3714.

2. 2560 MFCM Score Card Restriction

Punched cards scored on the column 80 end with the M-4 score are recommended for only reading in the 2560 MFCM. The M-4 score is not recommended when punching is required.

The 2560 "Machines" page in the Sales Manual will be changed to reflect this restriction.

OS/360 RELEASE 15- PLANNED IMPROVED ERROR RECOVERY PROCEDURES

The following planned changes for OS/360 Release 15 are provided for planning purposes. The actual availability and specifications will be as provided through the official IBM announcement and documentation procedures.

The error recovery procedures resident work area in the nucleus will be expanded from 400 bytes to 1024 bytes in Release 15. This will improve performance by requiring less loading of small individual sections of the ERP's. The number of re-reads will be reduced from 100 to 40. There will be no changes required of the user to his SYSGEN procedure to accommodate this change.

Users should evaluate their available storage to determine if they will have sufficient core to accomodate the 624 byte increase. If the user must reduce his storage allocation to accommodate this change, one possibility is to analyze the I/O specifications in light of what is actually needed. Elimination or reduction of such items will reduce the nucleus size needed.

OS/STRAM PTF'S AVAILABILITY

There are currently three PTF's (for IOS, BTAM and STRAM) active for OS/STRAM under Releases 12 and 13. They are not available from Field Engineering. Contact the OS/STRAM Coordinator at your district Field Systems Center for further information. Refer to STR PTF Numbers 3, 4 and 5. The PTF documentation supplies the detail as to the reasons for the fixes.

OS/360 Release 14 is now available and a new set of PTF's will be supplied for it.

NOTE: If the Field Systems Center has any difficulty in identifying these PTF's, the FSC Coordinator should call Mr. W. T. Romei, DPD Technical Center, Poughkeepsie, Tie Line 8-165-7876 or 914-485-7876.

OS/360 ATTEMPT TO USE UNSUPPORTED 2314-2844

The 2844 is not supported under OS/360. Therefore, careful consideration should be given to the implications involved before implementing the following change.

The change should be made to SYS1.GENLIB, the system macro IECIOS, at approximately statement number 56130000. The nearest preceding label is INT009. The instruction to change is:

```
Before   TM   UCBFL1 (UCBREG), UCBBSY
              Test for Busy UCB

         BC   8, INTATT      NO, UNSOLICITED
                             DE

After    TM   UCBFL1 (UCBREG), UCBBSY+
                               UCBASK

         BC   8, INTATT
```

OS/360 COBOL/ASSEMBLER TAPE READ ERROR RECOVERY ROUTINE

An OS/360 COBOL/Assembler Tape Read Error Recovery Routine has been provided to Field System Centers. The routines provide operator intervention and the recovery capability beyond what is provided by the Type I OS/360 Error Recovery Procedures. These routines have been found useful in solving user problems encountered with magnetic tape under OS/360.

The routines are documented in the Sales and Systems Guide Y20-0162-0, to be released from the IBM Distribution Center, Mechanicsburg. Refer to Publication Release Letters for specific availability. Do not order until it is so announced. The abstract of the manual follows:

The routines discussed in this manual are applicable only to tape read errors where BSAM or QSAM access methods are employed. In general, the routines can be used in any current operating system environment (as of September 1, 1967) with either COBOL or Assembler source programs, provided the operating system includes the SYNADAF and SYNADRLS (SVC 68) option.

This manual is intended for systems programmers. It discusses programming routines for extending tape error recovery procedures beyond those now available in an Operating System/360 environment where COBOL and Assembler are used. At present, after a tape read error, the system performs a series of steps that cause the tape drive to attempt to retransmit (up to a predetermined number of times) a physical record (block) correctly, and to record the event for future reference. In addition, the system has provisions for accommodating, via a SYNAD exit in the DCB, any user-written routines for handling such errors.

The routines discussed here may be considered as an extension of the system capability for handling tape read errors; as such, they may complement, supplement, or even replace any user-written routines; they are linked to the system and any user-written routines through the I/O Supervisor. A general description of the capabilities of these routines is given in the Preface of this manual.

OS/360 Release 14 FORTRAN-APPROACH TO CHANGED CSECT NAMING

OS/360 Release 14 was announced by Program Announcement P 68-7. The following is a suggested approach to the changed CSECT naming convention effective with Release 14.

Currently all OS/360 FORTRAN compilers develop the object module CSECT name by appending a pound sign (#) to the entry point name for that module. With Release 14, this convention will be changed so that the CSECT name and the entry point name will be identical for all modules compiled under Release 14 or later.

Because of this change, users maintaining libraries of FORTRAN object modules will now have both object modules with the pound sign in the CSECT name and those without it.

This change will affect all programming situations where FORTRAN object modules are referred to by CSECT name. The two most common situations are:

1) The programmer is creating an overlay structure.

2) The programmer is doing a CSECT replace to an existing structure.

In either case each object module must be referenced by CSECT name, and the programmer will have to know whether it contains a pound sign or not.

A possible approach to this situation is for the installation to maintain a list of current CSECT names and make this list available to the programmers.

A second approach is to apply the OPM's in SYS1. SAMPLIB which alter the compilers so that they again append the pound sign to CSECT names.

The following is a third approach, and one, which in many ways, is more satisfactory and easier to use than the other two. It requires no software or programming changes, and only minor modifications to control statements.

First, consider this approach as applied to overlay structures. The following modifications are required:

1) Placing two INSERT cards in the overlay deck for each object module, one card would have the CSECT name without pound sign and the other would have the CSECT name with it.

2) Indicating "LET" as an option in the PARM field of the linkage editor step execute card.

3) Indicating (8, LT, LKED) as an option in COND field of the go-step execute card.

If the CSECT being inserted was compiled on Release 14 or later, it will be inserted, and an unresolved external reference will be indicated for the INSERT card having the CSECT name with pound sign. The load module will, however, be marked executable because the "LET" option was specified. The unresolved external reference gives a severity code of 8, so the COND=(8, LT, LKED) is required to allow execution of the go-step.

If the CSECT being inserted was compiled on Release 13 or earlier, it will be inserted. In this case the CSECT name without pound sign (which appears on the second INSERT card) will be resolved to the entry point name, and no unresolved external reference will occur.

This approach can be easily adapted to use with standard catalog procedures, or if the user wishes, he may maintain a special catalog procedure to be used only with overlay structures.

Let us now consider this approach as applied to the CSECT replace situation. The only change here is that a "REPLACE" card must be included specifying the CSECT name with pound sign as the module to be replaced in the structure and specifying no new CSECT to replace it (i. e. REPLACE CSECT #). This "REPLACE" card and the new CSECT must appear in the job stream before the CSECT to be replaced.

The linkage-editor will look for the CSECT specified in the "REPLACE" card. If it finds the CSECT with pound sign, it will delete it. (It is deleted because no new CSECT was specified in the "REPLACE" card.) If it does not find the CSECT with pound sign, it ignores the "REPLACE" card.

If the CSECT to be replaced was compiled with Release 14 or later, its CSECT name will have no pound sign attached. Since the new CSECT appears first in the job stream, automatic replace will occur.

If the CSECT to be replaced was compiled with Release 13 or earlier, its CSECT name had a pound sign and would already have been deleted by the "REPLACE" card. The new CSECT would then be included in the structure.

This approach has the following advantages:

1) No necessity to maintain a list.

2) Programmers do not have to concern themselves with differences in CSECT names.

3) This approach should work indefinately with no further changes.

4) Installations will be gradually converting because they will be using new compilers which do not produce pound signs.

Installations which keep their libraries in load module form will still have to do a re-link-edit, but no re-compilation of FORTRAN H and early FORTRAN E programs is necessary with this approach.

NOTE:

This approach may allow jobs to get into the execute step with other unresolved external references from the link-edit step. This will be made apparent by the failure of the job to execute. The unresolved external references will be noted by linkage-editor warning messages.

OS/360 FORTRAN VARIABLE RECORD EXTENSION

The following is provided for planning purposes. The actual availability and specifications will be as announced through the formal IBM announcement and documentation procedures.

For variable length records written without format control, OS/360 FORTRAN permits a logical record to extend across more than one physical record or block. This facility is referred to as Variable Record Extension (VRE) and records of this type are referred to as "spanned records." Present plans are that the Operating System will not support VRE until Release 17. Therefore, care should be exercised in using this facility. Data sets containing spanned records intended as input to FORTRAN may not be handled by components other than FORTRAN. In addition, a utility may split a field so as to make the data unacceptable to FORTRAN.

INFORMATION ABOUT THE NEWSLETTER

EVERY SALESMAN AND SYSTEMS ENGINEER SHOULD RECEIVE ONE COPY OF THE NEWSLETTER. Each FE Branch Manager should receive five copies for distribution to Customer Engineers.

The IBM Installation Newsletter is distributed only to IBM locations and is not available to customers. The Newsletter is separated into two parts. The first part contains information which is intended only for IBM personnel and the pages are labeled For IBM Internal Use Only. THESE PAGES, OR REPRODUCTIONS OF THEM, ARE NOT TO BE GIVEN TO CUSTOMERS. However, using your own good judge-judgement, you may discuss pertinent information from these pages with a customer. The second part contains information which may be REPRODUCED by the Branch Office at their discretion and given to customers. These pages are not labeled with any restrictive classification. Note that when any abstracts for Type III programs are given to the customer, he must be informed that the disclaimer for Type III programs, as contained in the Newsletter, applies. YOUR COOPERATION IN PROPERLY HANDLING THE CONTENT OF THE IBM INSTALLATION NEWSLETTER IS NECESSARY FOR ITS CONTINUED PUBLICATION.

Changes to the quantity of the Newsletter shipped are to be made by the Administration Manager. (NOTE THAT EXTRA COPIES OF THE NEWSLETTER ARE NOT TO BE ORDERED FOR DISTRIBUTION OF INFORMATION TO CUSTOMERS.)

Direct inquiries concerning the contents of this Newsletter to:

W. P. Gaffigan, Editor
IBM Installation Newsletter
Installation Planning #713
DPD HQ
112 East Post Road
White Plains, New York 10601
Telephone 914-WH9-1900 Extension 4781

MARKETING PUBLICATIONS ANNOUNCED BY PRL'S

The weekly PRL's (Publications Release Letters) are used to insure that all Salesmen and Systems Engineers are aware of new or revised Marketing Publications. Normally, each issue of the Newsletter will contain information from two PRL's, one following the other. The information will be placed in the Newsletter in its original form with no rearrangement of form numbers or titles. It is not intended to replace existing information and distribution sources. You should be certain that you are aware of these sources.

-PRL #52 December 29, 1967-

This PRL was not available for previous publication

MARKETING PUBLICATIONS

FORM NO.	TITLE	DISPOSITION
520-1932-0	Linear Programming for Management Decisions	NEW
542-0034-1	CRA Announcement - Brochure	Use 542-0034-0
A24-3030-1	Message Rates for IBM 1448 Transmission Control Unit	Scrap A24-3030-0 & N24-0351
A24-3256-5	IBM 1285 Optical Reader Component Description and Operating Procedures	Scrap A24-3256-4
C20-1688-0	Operator Manual IBM 2260 Display Station, IBM 2848 Display Control, IBM 1053 Printer	NEW
C24-3337-2	IBM S/360 Operating System RPG	Use C24-3337-1, N26-0515, N26-0530 & N26-0546
C24-9000-2	IBM S/360 Model 20 Tape Programming System Control and Service Programs	Scrap C24-9000-1 & N24-9010-0
C24-9009-1	IBM S/360 Mod 20 Tape Programming System Operating Procedures	Scrap C24-9009-0
C26-3709-1	IBM 1130 Disk Monitor System, Version 2 System Introduction	Scrap C26-3709-0
C26-3715-1	IBM 1130/1800 Basic FORTRAN IV Language	Scrap C26-3715-0
C26-3808-2	IBM S/360 Mod 20 Tape Programming System Utility Programs	Scrap C26-3808-1 & N33-9009
C26-5927-3	IBM 1130 Assembler Language	Scrap C26-5927-2
C26-5929-3	IBM 1130 Subroutine Library	Scrap C26-5929-2, N26-0551 & N26-0553
H20-0478-0	IBM S/360 Model 20 Bill of Material Processor Application Description	NEW
N20-1066	TNL to 1130 Project Control System (1130-CP-05X) Operator's Manual Re: H20-0343-1	NEW
N23-0611	ENL IBM S/360 DOS/TOS Language Interface Guide Re: R20-1057-0	NEW
N28-3001	TNL - IBM S/360 Time Sharing System Linkage Editor Re: C28-2005-1	NEW
N28-3002	TNL IBM S/360 Time Sharing System Independent Utilities Re: C28-2038-0	NEW
N28-3004	TNL IBM S/360 Time Sharing System Command Language for Manager's and Administrator's Re: C28-2024-0	NEW
N28-3008	TNL IBM S/360 Time Sharing System System Programmer's Guide Re: C28-2008-0	NEW
N33-8518	TNL to IBM S/360 Mod 20 Tape Programming System, Performance Estimates Re: C24-9010-0 with N33-9005	NEW
N33-8519	TNL to S/360 Mod 20 TPS Sort/Merge Re: C26-3804-1	NEW
R29-0094-0	IBM S/360 Mod 20 RPG Coding P.I. Examination	NEW
R29-0249-1	IBM CSF P.I. Examination	Use R29-0249-0
Y28-0242-1	IBM S/360 Time Sharing System On-Line Test Control Program PLM	Scrap Y28-2042-0
Y28-3058-0	IBM S/360 Time Sharing System Assembler Program Logic Manual Re: Y28-2021-0	NEW
Y28-3060-0	IBM S/360 Time Sharing System - Independent Utilities PLM Re: Y28-2039-0	NEW
Z20-1856-0	1130 Corrugator Scheduling Program	NEW
Z32-0394-0	ENL IBM Comp. System Training Education Guide Re: Z32-0401-1	NEW
Z77-7274-0	DOS Multiprogramming Demonstration with Job Stream on Disk	NEW
Z77-7275-0	IBM 1401 Supplemental Study Course	NEW
Z77-7276-0	Solving Transportation Problems Using Network Flow Theory	NEW
Z77-7277-0	The Installation of an IBM System/360 Model 20 Equipped with a Communications Adapter	NEW

- PRL #2 January 12, 1967 -

C20-1619-4	Catalog of Programs for IBM System/360 - December 1967	Scrap C20-1619-3 & N20-0(
N20-1130.25	1130 SRL Newsletter Re: A26-5916-2	Scrap N20-1130.24

MARKETING PUBLICATIONS

221-0461-0	The IBM 1800 Data Acquisition and Control System	NEW
221-0462-0	The IBM 1800 Multiprogramming Executive (MPX) Operating System	NEW
221-0463-0	The IBM 1800 MPX Operating System I/O Control Routines (IOCR)	NEW
221-0464-0	The IBM 1800 MPX Operating System Executive Director	NEW
221-0465-0	The IBM 1800 MPX Operatang System Batch Processing Monitor	NEW
320-1906-11	In Brief, November 1967	Scrap 320-1906-10
543-0033-2	Selectric I/O Manual	Scrap 543-0033-1
520-1551-0	1130 Numerical Surface Techniques & Contour Map Plotting	NEW
520-1914-0	VSP/360: Direct Route to Improved Vehicle Scheduling	NEW
520-2062-0	The IBM 633 Billing System	NEW

PRL #2 (Continued)

A24-3264-1	1030 Original Equipment Manufacturer's Information	Use A24-3264-0 & N24-022
A32-0001-0	IBM Tape Drive, 2400 Series Model 6, Tape Requirements	NEW
C24-1456-2	Sort Timing Specs & Operating Procedures for IBM 1401/1460	Use C24-1456-1, N24-0281 N21-5002
C24-3427-2	IBM S/360 DOS Data Management Concepts	Use C24-3427-1, N24-5122 5169, N24-5197, N24-5276 N24-5291
C24-5030-3	IBM S/360 Disk & Tape Operating Systems Concepts & Facilities	Use C24-5030-2, N24-5182 5191, N24-5233, N24-5275 N24-5340
C26-3751-1	IBM 1800 Card/Paper Tape Programming System Operator's Guide	Scrap C26-3751-0, N26-05 N26-0528, N26-0539 & N2
C26-5880-4	IBM 1800 Subroutine Library	Scrap C26-5880-3 & N26-05
C27-6937-0	IBM S/360 Operating System and 1130 Disk Monitor System FORTRAN IV Subroutines for Data Transmission Between a System/360 and an 1130 System	NEW
C27-6938-0	IBM S/360 Operating System and 1130 Disk Monitor System: User's Guide for Job Control From an IBM 2250 Display Unit Attached to an IBM 1130 System Preliminary Specifications	NEW
C28-6514-5	IBM S/360 Operating System, Assembler Language	Scrap C28-6514-4, N26-05 N28-2150
C28-6632-2	IBM S/360 Operating System: Job Control Language Charts	Use C28-6632-1 & N28-225
C28-6647-1	IBM S/360 Operating System: Supervisor & Data Management Macro-Instruction	Use C28-6647-0, N28-2217 2223, N28-2248 & N28-225
C28-6673-0	IBM S/360 Operating System: Planning for the Shared Direct Access Device Option	NEW
E20-0299-0	Computer Application in Nuclear Medicine	NEW
E50-0036-0	Using the IBM 1232 Optical Mark Page Reader in the Federal Government	NEW
H20-0136-2	Mathematical Programming System/360 (360A-CO-14X) Application Description	Scrap H20-0136-1 & N20-1C
H20-0237-3	Structural Engineering System Solver (STRESS) for the IBM 1130, Model 2B (1130-EC-03X) Version II Application Program Bulletin	Scrap H20-0237-2
H20-0323-2	Attached Support Processor System (ASP) (360A-CX-15X) System Programmer's Manual	Scrap H20-0323-1
H20-0340-2	Structural Engineering System Solver (STRESS) for the IBM 1130, Model 2B (1130-EC-03X) Version II User's Manual	Scrap H20-0340-1
H20-0374-1	System/360 AUTOSPOT (360A-CN-08X) and AD-APT/AUTOSPOT (360A-CN-09X) Numerical Control Processors - Version 2 Operator's Manual	Scrap H20-0374-0
H20-0476-0	Mathematical Programming System/360 (360A-CO-14X) Linear and Separable Programming User's Manual	NEW
J20-0009-1	QUIKTRAN Financial Programs	Use J20-0009-0
K20-0042-1	Construction Project Management Control System at the H. B. Zachry Company	NEW - REVISED ABSTRACT
K20-0247-0	BUFFUMS - Unit Control Using the IBM System/360	NEW
N20-1065	TNL for System/360 AD-APT/AUTOSPOT Numerical Control Processor (360A-CN-09X) Version 2 - Application Description Re: H20-0463-0	NEW
N26-0578	IBM 1800 Time-Sharing Executive System Operating Procedures Re: C26-3754-2 with N26-0561, N26-0569 & N26-0571	NEW
N28-0242	IBM 7090/7094 IBSYS Operating System, Version 13: System Monitor (IBSYS) Re: C28-6248-3, -4, -5, -6, -7 with N28-0207, N28-0182, N28-0187, N28-0137, N28-0166 & N28-0139	NEW
N28-3000	TNL IBM S/360 Time Sharing System Assembler Language Re: C28-2000-2	NEW
N28-3003	TNL IBM S/360 Time Sharing System Command Language User's Guide Re: C28-2001-1	NEW
N28-3009	IBM S/360 Time Sharing System System Generation and Maintenance Re: C28-2010-0	NEW
N33-8000	IBM S/360 Operating System ALGOL Programmer's Guide Re: C33-4000	NEW
R20-4095-0	IBM Branch Office Library Education Guide	NEW
R20-9172-0	IBM S/360 for Data Processing Managers Course Description	NEW
R29-0030-3	IBM FORTRAN P.I. Advisor Guide	Use R29-0030-2
R29-0085-3	FORTRAN IV for IBM S/360 P.I. Illustrations	Use R29-0085-2
R29-0093-2	IBM S/360 RPG Coding (Using 2560 MFCM) P.I. Illustrations	Use R29-0093-1 & N23-0609
R29-0096-2	IBM S/360 RPG Coding P.I. Text	Use R29-0096-1
R29-0106-1	IBM 1130 FORTRAN P.I. Problem Book	Use R29-0106-0
R29-0110-1	IBM S/360 PL/I Coding P.I. Advisor Guide Supplement	Scrap R29-0110-0
R29-0162-1	IBM PCDPP 514 Oper. P.I. Notebook	Use R29-0162-0
R29-0206-2	IBM S/360 COBOL - COBOL Prog. Fund. P.I. Reference Handbook	Use R29-0206-1
R29-0241-2	IBM CSF P.I. Text - Unit 1	Use R29-0241-1
R29-0248-3	IBM CSF P.I. Notebook 2-6	Use R29-0248-2
V25-6465-0	IBM S/360 DOS Exercises Card Deck	New - Use R20-4074-0
Y20-0081-1	Structural Engineering System Solver (STRESS) for the 1130, Model 2B (1130-EC-03X) Version II System Manual	Scrap Y20-0081-0
Y20-0147-0	TNL to 1130 Linear Programming Mathematical Optimization Subroutine System (1130-LP-MOSS) (1130-CO-16X) System Manual Re: Y20-0141-0	NEW

PRL #2 (Continued)

Y20-0153-0	Medical Information System Programs (MISP) (360A-UH-08L) Application Description	NEW
Y24-5070	TNL to S/360 DOS System Control PLM Re: Y24-5017-2 with Y24-5059	NEW
Y24-5077	TNL to IBM S/360 BOS Logical IOCS PLM Re: Y24-5003-1	NEW
Y28-3050	TNL to IBM S/360 Time Sharing System Task Monitor PLM Re: Y28-2041	NEW
Y28-3059-0	IBM S/360 Time Sharing System Linkage Editor PLM Re: Y28-2030-0	NEW
Y28-6605-3	IBM S/360 Operating System: Introduction to Control Program Logic, PLM	Use Y28-6605-2, Y28-2180, Y28-2203, Y28-2222, Y28-2261 & Y28-2219
Z04-7633-1	Five Minute Typing Test & Copying From Rough Draft	Use M04-7633-0
Z20-0839-1	IBM 1050 Data Communications Dial-up Systems Design	Use Z20-0839-0
Z77-6025-1	Queueing Analysis of Real-Time Computer Programming	Scrap Z77-6025-0
Z77-7231-1	1050 Configurator	Scrap Z77-7231-0
Z77-7266-1	The Customer Information File	Use Z77-7266-0
Z77-7282-0	RCA 301 to System/360 Marketing Assistance Guide	NEW
Z77-7284-0	PSM with OS/360 SORT	NEW
Z77-7285-0	System/360 APT N/C System Implementation Version I Post Processor Conversion	NEW
Z77-7286-0	Remote Job Entry & Output for Disk Operating System (DROS)	NEW
Z77-7288-0	Shared File Operation under DOS using the 2314	NEW

- PRL #3 January 19, 1968-

C20-1602-7	Catalog of Programs for IBM 705-1410-7010-7070-7074-7080-7740 and 7750 DPS - Dec. 1967	Scrap C20-1602-6 & N20-0014-11
C20-1603-7	Catalog of Programs for IBM 1620 and 1710 DPS - December 1967	Scrap C20-1603-6 & N20-0015-12
C20-1604-7	Catalog of Programs for IBM 704-709-7040-7044-7090 and 7094 DPS - December, 1967	Scrap C20-1604-6 & N20-0016-12
C20-1630-3	Catalog of Programs for IBM 1130 Computer System and IBM 1800 Data Acquisition and Control System - December 1967	Scrap C20-1630-2 & N20-0031-12
N20-1440.49	1440 SRL Newsletter Re: A24-3005-5	Scrap N20-1440.48
N20-7040.39	7040 SRL Newsletter Re: A28-6288-4	Scrap N20-7040.38
Z20-0305-16	KWIC Index to TIE December 1967	Scrap Z20-0305-15, Z20-1858-0 & Z20-1862-0

MARKETING PUBLICATIONS

221-0400-3	The IBM 1442 Card Punch, Model 5 Proposal Insert	Use 221-0400-2
520-1077-1	Pencil, Paper and the IBM 1231/1232 Optical Mark Page Reader Promotional Brochure	NEW - REINSTATED
520-2060-0	IBM 1030 Data Collection System for Manufacturing Industries	NEW
520-2061-0	IBM 1030 Demonstration Kit Pocket Folder	NEW
546-0030-0	IBM Photo Master Papger Folder	NEW
570-0295-2	IBM One-Half Inch Magnetic Tape Specs for use at 556, 800 bpi and 3200 fci	Scrap 570-0295-1
A24-3333-4	IBM 59 Card Verifier Reference Manual	Scrap A24-3333-3
A26-5917-6	IBM 1130 System Summary	Scrap A26-5917-5
C24-3374-2	IBM BPS RPG (Card) Specifications SRL Bulletin	Use C24-3374-1
C24-9003-2	IBM S/360 Model 20 Tape Programming System I/O Control System	Scrap C24-9003-1, N33-9004 & N33-8508
H20-0290-2	Mathematical Programming System/360 (360A-CO-14X) Control Language User's Manual	Scrap H20-0290-1
H20-0373-1	System/360 AUTOSPOT Numerical Control Processor (360A-CN-08X) Version 2 - Part Programming Manual	Scrap H20-0373-0
H20-0375-1	System/360 AD-APT/AUTOSPOT Numerical Control Processor (360A-CN-09X) Version 2 - Part Programming Manual	Scrap H20-0375-0
H20-0483-0	Advanced Life Information System Policy Master Record Code Book	NEW
J20-0012-1	QUIKTRAN Continuous Beam Analysis Program	Scrap J20-0012-0
K20-0231-0	Data Processing at High School District 214 Mount Prospect, Illinois	NEW
K20-0241-0	IBM 1130 Computing System-Policy Preparation and Premium Accounting-Morley, Watson and Baldwin Incorporated	NEW
K20-0245-0	Production Planning and Control System at the Minster Machine Company	NEW
N27-1292	TNL to IBM S/360 Operating System: Multiprogramming with a Fixed Number of Tasks (MFT): Concepts and Considerations Re: C27-6926-0	NEW
N27-2916	TNL IBM S/360 Model 65 Functional Characteristics Re: A22-6884-2 with N27-2909	NEW
N27-2918	IBM 1130 Computing System Component Description IBM 2250 Display Unit Model 4 TNL Re: A27-2723-0 with N27-2915	NEW
N28-0564	IBM S/360 FORTRAN IV Library Subprograms Re: C28-6596-2	NEW
N28-3012	IBM S/360 Time Sharing System Assembler Programmer's Guide Re: C28-2032-0	NEW

PRL #3 (Continued)

R20-8026-0	IBM 1130 Comp. System Assembler Language Coding Education Guide	NEW
R29-0097-2	IBM S/360 P.I. RPG Coding Illustrations	Use R29-0097-1
R29-0100-1	FORTRAN for the IBM 1130 P.I. Advisor Guide	Use R29-0100-0
R29-0243-1	IBM CSF P.I. Text - Unit 2	Use R29-0243-0
R29-0244-2	IBM Computing System Fundamentals P.I. Text - Unit 3	Use R29-0244-1
R29-0245-2	IBM Computing System Fundamentals P.I. Text - Unit 4	Use R29-0245-1
R29-0246-1	IBM Computing System Fundamentals P.I. Text - Unit 5	Use R29-0246-0
X20-1748-2	DOS/TOS Job Control Language & Operator Communication Reference Card	Scrap X20-1748-1
Y20-0140-0	TNL to Mathematical Programming System (360A-CO-14X) System Manual Re: Y20-0065-0	NEW
Y20-0149	TNL for System/360 AUTOSPOT Numerical Control Processor (360A-CN-08X) Version 2 - System Manual Re: Y20-0119-0	NEW
Y24-5073	TNL to S/360 TOS System Control PLM 1287 & Simpl. Tape Labels Re: Y24-5022-1 with Y24-5060 & Y24-5063	NEW
Y28-2011-1	IBM S/360 Time Sharing System System Control Blocks PLM	Scrap Y28-2011-0
Y28-2015-1	IBM S/360 Time Sharing System System Generation & Maintenance PLM	Scrap Y28-2015-0
Y28-3057	IBM S/360 Time Sharing System FORTRAN IV PLM TNL Re: Y28-2019-0	NEW
Z77-7289	The Job Accounting System - A Tool to Measure Job Shop Computer Systems	NEW
Z77-7290	A Selected Bibliography for Operations Research in the Product Industry	NEW

February 9, 1968 Issue No. 68-02

IBM INSTALLATION NEWSLETTER
TABLE OF CONTENTS

OS/360 RELEASE 14-SYSTEM GENERATION CHANGES CORRECTION

Installation Newsletter issue 67-26 contained, on page 2, an article on the above subject.

The coding shown in item 1 is in error. The fifth line in the coding should be changed from // SYSLMOB to // SYSLMOD. Following this entry should be a line of coding:

```
// MODLIB  DD   DSNAME = SYS1.MODLIB,
                        DISP = OLD
```

then // SYSLIN, etc.

The second sentence in the last paragraph of the article contains a transposition. Change IBECOPY to IEBCOPY.

OS/360 NEW JCL EXAMPLES REFERENCE CARD

A new reference card (X20-1760) is available for OS/360 Job Control Language. The card explains, by example, how to prepare JCL for FORTRAN, COBOL, PL/I, Assembler, and Utilities. It represents a new approach to detailed explanation of Job Control Language.

The card is available from the IBM Distribution Center, Mechanicsburg.

OS/360 MVT-BUFNO IN IEFDATA CARD CONSIDERATIONS

The reader-interpreter builds a JFCB in the job queue for each DD* data set. The skeleton comes from the IEFDATA DD card. If BUFNO is specified, a BSAM open will get buffers that take space which will not be used. The COBOL F compiler reads SYSIN data with BSAM. Blocked input with BUFNO = 1 caused a 100K region to be small, although SIZE = 10000 and BUF = 20000 had been specified. The compiler gets its own buffers and does not use those provided by the system at OPEN time. Thus, all buffers obtained at OPEN time, in response to the BUFNO parameter, are unused and may take up space required for other processing. It is recommended that BUFNO be omitted in the DCB parameters on the IEFDATA card.

The IEFDATA DCB parameters are used both during the SPOOL function and the later read. Therefore, the reader using QSAM will obtain the default of 2 buffers if BUFNO is omitted. This will necessitate expansion of the reader region size when blocking is being employed, but should increase the efficiency of the reader.

OS/360 DATA CELL RELATIVE TRACK TO ABSOLUTE CONVERSION

The following is a field contribution which has not been submitted to any formal IBM test. Potential users are expected to evaluate its usefulness in their own environment.

This routine, shown in Exhibit 1, was written to solve the problem of addressing a Data-Set which extends over more than 64,000 tracks. While OS/360 offers the user relative block addressing in the form 'BBB', the three byte block address is first converted to a relative track address of the form 'TTRZ' thus truncating the track address to a two byte value. This routine has proven helpful in addressing within one data cell. Multiple DCB's and problem program switching are required to handle more than one data cell.

1. Restrictions

   As written, this routine converts only a relative track address and assumes full track records. The record portion of the resulting 'MBBCCHHR' is always returned as record one.

2. Departures from OS/360 Conventions

   In the interest of speed, registers 0, 1 and 2 are used as parameter registers rather than passing a parameter list address.

   All registers are saved at entry to the program but since this is not a calling program, no save area is provided.

   Register 15 is used as a base register and must contain the entry point address upon entering the module.

3. Entry to Routine

   The following conditions are required at entry:

   a. Register 0 must contain the relative track number right justified. The maximum decimal value is 16,777,215.

b. Register 1 must contain the address of the opened DCB for the Data-Set.

c. Register 2 must contain the address of an 8 byte area into which the absolute address 'MBBCCHHR' will be formatted. No alignment is necessary.

d. Register 14 must contain the return address to the calling program.

e. Register 15 must contain the entry address of the routine.

4. Program Logic

a. The DEB address is obtained from the DCB.

b. Using the DEBNMEXT value as a counter, Register 0 is decremented by the number of tracks in each extent until it goes minus (extent found) or all extents are exhausted (track not in extents).

c. 'M' is developed and placed in the receiving area.

d. 'BB' is moved into the receiving area from the DEB.

e. 'R' is moved in as a constant of one.

f. The extent start address (DEBSTRCC) is converted to relative track and added into Register 0.

g. The resulting relative address is converted to absolute and stored in "CCHH" of the receiving area.

h. Return is made to caller.

5. Exit from Routine

a. Registers 0 through 14 are restored.

b. Register 15 contains:
0 if extent was found.
4 if extent not found.

6. Comments

It may be noted that the table with symbolic name 'MYCCHH' which is used as both a multiplier and a divisor, contains a value of 1 in the last half word. This results in a 'NO-OP' in this particular application, but gives the facility to convert any 'CCHH' combination by simply changing the values in the table.

OS/360 COBOL F BLOCK SIZE OVERRIDE

When the block size for a file is specified as "BLOCK CONTAINS 0 RECORDS" in the file section of a COBOL source program, the block size can be overriden at object time. This can be done by the user specifying BLOCK SIZE for that file in the DCB parameter of the DD card. The ability to override block size can be used for files that consist of fixed length records.

When QSAM and QISAM are used as access methods, the override facility can be used for multiple files and no further changes to the COBOL source program are required. This is because QSAM and QISAM use dynamic buffering.

When BSAM or BISAM are used as access methods, the DCB override facility can be used for only one file. The buffer is located in the object module. A level 77 filler must be inserted at the beginning of working storage to reserve core for the largest expected block size. This technique will enable the first file named in the file section in the program to have its block size specified at object time.

OS/360 RELEASE 14-FORTRAN H NEW XREF OPTION

The XREF option is new with Version II of FORTRAN H. When using this option, a DD card is required in the compile step for SYSUT2. This information appears on Page 12 of the FORTRAN H Program Logic Manual (Y28-6642).

OS/360 DATA CELL RELATIVE TRACK TO ABSOLUTE CONVERSION

```
 LOC  OBJECT CODE     ADDR1 ADDR2  STMT   SOURCE STATEMENT                                F01JUN67   1/12/68

000000                              2 CONVTTT  CSECT
                                    3 * *  ROUTINE TO CONVERT RELATIVE TRACK 'TTT' TO ABSOLUTE 'MBBCCHHR'.


                                    5 *      REGISTER CONTENTS AT ENTRY: 0=TTT
                                    6 *                                  1=DCB POINTER.
                                    7 *                                  2=8-BYTE AREA FOR 'MBBCCHHR'.
                                    8 *                                 14=RETURN ADDRESS TO CALLER.
                                    9 *                                 15=ENTRY ADDRESS USED AS BASE.

                                   11 *      RETURN CONDITIONS: REGISTER 15=0, NORMAL RETURN.
                                   12 *                         REGISTER 15=4, TRACK NOT IN EXTENTS.

 LOC  OBJECT CODE     ADDR1 ADDR2  STMT   SOURCE STATEMENT

                                   14           SAVE  (14,12),,*
000000 47F0 F00C             0000C 15+          B     12(0,15) BRANCH AROUND ID
000004 07                          16+          DC    AL1(7)
000005 C3D6D5E5E3E3E3              17+          DC    CL7'CONVTTT' IDENTIFIER
00000C 90EC D00C             0000C 18+          STM   14,12,12(13) SAVE REGISTERS
000000                             19           USING CONVTTT,BASE
000000                             20           USING IHADCB,DCBREG
000000                             21           USING IHADEB,DEBREG
000010 5811 002C             0002C 22           L     DEBREG,44(DCBREG)   LOAD BASE FOR DEB.
000014 1B55                        23           SR    BCTREG,BCTREG       ZERO BCT REGISTER.
000016 4350 1010             00010 24           IC    BCTREG,DEBNMEXT     LOAD NUMBER OF EXTENTS.
00001A 1865                        25           LR    EXTREG,BCTREG       SAVE FOR DEVELOPING 'M'.
00001C 4830 102E             0002E 26 CNVTB     LH    WORK1,DEBNMTRK      LOAD NUMBER OF TRACKS IN EXTENT.
000020 1B03                        27           SR    TTTREG,WORK1        SUBTRACT FROM RELATIVE TTT.
000022 4740 F03C             0003C 28           BM    CNVTC               IF TTT GOES MINUS, EXTENT FOUND.
000026 4111 0010             00010 29           LA    DEBREG,16(DEBREG)   BUMP TO NEXT EXTENT.
00002A 4650 F01C             0001C 30           BCT   BCTREG,CNVTB        BRANCH IF STILL IN DEB.
00002E 4150 0004             00004 31           LA    BCTREG,4            DEVELOP ERROR CODE.
000032 505D 0010             00010 32 RETURN    ST    BCTREG,16(13)       STORE CODE IN SAVE AREA.
                                   33           RETURN (14,12)            TETURN TO CALLER.
000036 98EC D00C             0000C 34+          LM    14,12,12(13) RESTORE THE REGISTERS
00003A 07FE                        35+          BR    14 RETURN
```

Exhibit 1

```
00003C 1865                                37 CNVTC    SR    EXTREG,BCTREG        DEVELOP 'M'.
00003E 4262 0000              00000        38          STC   EXTREG,0(MBCHREG)   STORE 'M' IN MBBCCHHR.
000042 D205 2001 1024 00001   00024        39          MVC   1(6,MBCHREG),DEBBINUM MOVE BBCCHH TO MBBCCHHR.
000048 9201 2007      00007                40          MVI   7(MBCHREG),X'01'     FORCE RECORD 1 IN MBBCCHHR.
00004C 4A00 102E              0002E        41          AH    TTTREG,DEBNMTRK      ADD BACK OVERDRAW.
000050 4770 F05C              0005C        42          BNZ   CNVETTT              IF NET IS ZERO, NO CONVERT NEEDED.
000054 4150 0000              00000        43 NORMAL   LA    BCTREG,0             DEVELOP NORMAL RETURN CODE.
000058 47F0 F032              00032        44          B     RETURN               RETURN TO CALLER.
                                           46 * *  THE FOLLOWING CODE CONVERTS THE EXTENT START ADDRESS TO TTT.

00005C 4150 0004              00004        48 CNVETTT  LA    BCTREG,4             LOAD BCTREG FOR THREE TIME LOOP.
000060 4110 1026              00026        49          LA    DEBREG,DEBSTRCC      LOAD ADDRESS OF CCHH IN DEB EXTENT.
000064 4140 F0AA              000AA        50          LA    TBLREG,MYCCHH        LOAD ADDRESS OF MULTIPLIER TABLE.
000068 1B33                                51 CNVLOOP1 SR    WORK1,WORK1          ZERO IC REGISTER.
00006A 4331 0000              00000        52          IC    WORK1,0(DEBREG)      INSERT IN ORDER-C1,C2,H1,H2.
00006E 4C34 0000              00000        53          MH    WORK1,0(TBLREG)      MULTIPLY BY MYCCHH CONSTANTS.
000072 1A03                                54          AR    TTTREG,WORK1         ACCUMULATE TTT IN EXTENT.
000074 4111 0001              00001        55          LA    DEBREG,1(DEBREG)     BUMP TO NEXT CCHH IN DEB.
000078 4144 0002              00002        56          LA    TBLREG,2(TBLREG)     BUMP TO NEXT MULTIPLIER.
00007C 4650 F068              00068        57          BCT   BCTREG,CNVLOOP1      LOOP 3 TIMES.


                                           59 * *  THE FOLLOWING CODE CONVERTS TOTAL TTT BACK TO CCHH.

000080 4150 0003              00003        61          LA    BCTREG,3             LOAD BCTREG FOR TWO TIME LOOP.
000084 4140 F0AA              000AA        62          LA    TBLREG,MYCCHH        LOAD ADDRESS OF DIVIDE TABLE.
000088 8C00 0020              00020        63 CNVLOOP2 SRDL  TTTREG,32            SHIFT REMAINDERS.

  LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT

00008C 4864 0000              00000        64          LH    EXTREG,0(TBLREG)     LOAD DIVISOR.
000090 1D06                                65          DR    TTTREG,EXTREG        DIVIDE.
000092 4212 0003              00003        66          STC   QUOTREG,3(MBCHREG)   STORE C1,C2,H1, IN ORDER.
000096 4122 0001              00001        67          LA    MBCHREG,1(MBCHREG)   BUMP MBBCCHHR POINTER.
00009A 4144 0002              00002        68          LA    TBLREG,2(TBLREG)     BUMP DIVISOR POINTER.
00009E 4650 F088              00088        69          BCT   BCTREG,CNVLOOP2      LOOP TWICE.
0000A2 4202 0003              00003        70          STC   TTTREG,3(MBCHREG)    STORE H2 IN MBBCCHHR.
0000A6 47F0 F054              00054        71          B     NORMAL               RETURN TO CALLER - NORMAL.
                                           73 * *  CONSTANTS FOR CONVTTT.

0000AA 03E8                                75 MYCCHH   DC    H'1000'              C1 MULTIPLIER.
0000AC 0064                                76          DC     H'100'              C2 MULTIPLIER.
0000AE 0014                                77          DC    H'20'                H1 MULTIPLIER.
0000B0 0001                                78          DC    H'1'                 H2 MULTIPLIER.
```

Exhibit 1 (Continued)

February 9, 1968 Issue No. 68-02

```
000001                          80 DEBREG   EQU   1
000001                          81 DCBREG   EQU   1
000004                          82 TBLREG   EQU   4
00000F                          83 BASE     EQU   15
000005                          84 BCTREG   EQU   5
000006                          85 EXTREG   EQU   6
000003                          86 WORK1    EQU   3
000004                          87 WORK2    EQU   4
000000                          88 TTTREG   EQU   0
00000E                          89 RETREG   EQU   14
000002                          90 MBCHREG  EQU   2
000001                          91 QUOTREG  EQU   1
  LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT

                                93 * * DUMMY SECTIONS FOR ADDRESSING DEB'S AND DCB'S.

                                95 * *  DUMMY SECTION FOR DCB.

000000                          97 IHADCB   DSECT
000000                          98         DS    44C
00002C                          99 DCBDEBAD DS    F


                               101 * *  DUMMY SECTION FOR DEB.

000000                         103 IHADEB   DSECT
000000                         104         DS    F
000004                         105 DEBAMIND DS    C
000005                         106         DS    11C
000010                         107 DEBNMEXT DS    C
000011                         108         DS    15C
000020                         109 DEBDVMOD DS    F
000024                         110 DEBBINUM DS    H
000026                         111 DEBSTRCC DS    H
000028                         112 DEBSTRHH DS    H
00002A                         113 DEBENDCC DS    H
00002C                         114 DEBENDHH DS    H
00002E                         115 DEBNMTRK DS    H
000000                         116         END   CONVTTT
```

Exhibit 1 (Continued)

OS/360 FORTRAN H VERSION II-MINIMUM COMPILER SIZE

FORTRAN H (Version II) can be SYSGEN'ed for a minimum compiler size of 153,600 Bytes of storage. This size compiler will handle about 200 to 300 source cards. Each additional 18,000 bytes will allow roughly 100 additional source cards.

A significant point to note is that the SYSGEN default for compiler size is 204,800 bytes in order to get the minimum size FORTRAN H compiler, that size (153,600) must be explicitly specified in the FORTRAN SYSGEN macro.

OS/360 PL/I ARRAY LOWER BOUND

The lower bound of an array is 1 unless explicitly declared otherwise. A common error is to use a subscript of Ø in an array reference. This error may be difficult to isolate if the SUBSCRIPTRANGE condition is not enabled for program checkout.

OS/360 PL/I AL ROUTINE TO DETERMINE CORE REQUIRED

The following is a field contribution which has not been submitted to any formal IBM test. Potential users are expected to evaluate its usefulness in their own environment.

The following is an Assembly Language subroutine which, when called from a PL/I program, gives the amount of core required at that time. It will assist a PL/I oriented installation in determining where to set partition sizes or when a reduction in program size should be considered. The subroutine (Exhibit 1) is called from a PL/I program. This approach has been tested (by the author) in an MFT environment.

OS/360 PL/I AL ROUTINE TO DETERMINE CORE REQUIRED

```
//JOB65  JOB  MSGLEVEL=1
//C  EXEC  PROC=ASMFC,PARM.ASM='NODECK'
//ASM       EXEC PGM=IEUASM,REGION=50K                                  00020000
//SYSLIB DD     DSNAME=SYS1.MACLIB,DISP=SHR                            00040000
//ASM.SYSUT1  DD  UNIT=2314
//SYSUT1 DD     UNIT=SYSSQ,SPACE=(1700,(400,50))                       00060000
//ASM.SYSUT2  DD  UNIT=2314
//SYSUT2 DD     UNIT=SYSSQ,SPACE=(1700,(400,50))                       00080000
//ASM.SYSUT3  DD  UNIT=2314
//SYSUT3 DD     UNIT=(SYSSQ,SEP=(SYSUT2,SYSUT1,SYSLIB)),              X00100000
//              SPACE=(1700,(400,50))                                   00120000
//SYSPRINT DD   SYSOUT=A                                               00140000
//SYSPUNCH DD   SYSOUT=B                                               00160000
//ASM.SYSIN          DD UNIT=2314,VOLUME=SER=SYSLIB,                  C
//              DISP=(OLD,DELETE),DSNAME=D00.JOB65
IEF236I ALLOC. FOR JOB65     ASM       C
IEF237I SYSLIB   ON 134
IEF237I SYSUT1   ON 135
IEF237I SYSUT2   ON 136
IEF237I SYSUT3   ON 137
IEF237I SYSPRINT ON 130
IEF237I SYSPRINT ON 132
IEF237I SYSPUNCH ON 130
IEF237I SYSPUNCH ON 132
IEF237I SYSIN    ON 134
```

Exhibit 1

PROGRAM TO DETERMINE CORE USED

F01JUN67 12/27/67

```
LOC    OBJECT CODE     ADDR1 ADDR2  STMT   SOURCE STATEMENT
                                      2 *
                                      3 *
                                      4 *        PL/I CALLING SEQUENCE -- CALL FQEMAP;
                                      5 *
                                      6 *
                                      7 *        SIZE OF PROBLEM PROGRAM AT TIME OF CALL WILL BE TYPED
                                      8 *        ON CONSOLE.
                                      9 *        IT IS ADVISABLE TO INSERT SEVERAL CALLS IN A PROGRAM
                                     10 *        SINCE ITS SIZE MAY CHANGE DYNAMICALLY.
                                     11 *        TJK,362,12/28/67
                                     12 *
000000                               13 FQEMAP   START     0
000000                               14          USING     *,15
000000 47F0 F00C             0000C   15 STARTGR  B         G01
000004 06                            16          DC        AL1(6)
000005 C6D8C5D4C1D7                  17          DC        C'FQEMAP'
00000B 00
00000C 90EB D00C             0000C   18 G01      STM       14,11,12(13)
                                     19          DROP      15
000010 05A0                          20          BALR      10,0  BASE REG
000012                               21          USING     *,10
000012 4100 0048             00048   22          LA        0,72
000016 58F0 A0EE             00100   23          L         15,ADDR1
00001A 05EF                          24          BALR      14,15  GET DYNAMIC SAVE AREA OF 72 BYTES
00001C 9280 D000       00000         25          MVI       0(13),X'80'
                                     26 *        SET PICA FOR THIS ROUTINE
                                     27          SPIE
000020 0700                          28+         CNOP  2,4
000022 4110 A01C             0002E   29+         LA    1,*+12 LOAD BRANCH ADDRESS
000026 0511                          30+         BALR  1,1 BRANCH AROUND PARAMS.
000028 00000000                      31+         DC    A(0)· EXIT ROUTINE ADDRESS
00002C 0000                          32+         DC    AL2(0) INTERUPTION MASK
00002E 0A0E                          33+         SVC   14 ISSUE SPIE SVC
000030 5010 A0E6             000F8   34          ST        1,HOLD  SAVE ADDRESS OF PREVIOUS PICA
                                     35 *
000034 4150 0010             00010   36          LA        5,16  LD ADDR OF PTR TO CVT
000038 5855 0000             00000   37          L         5,0(5)  LD PTR TO CVT
00003C 5855 0000             00000   38          L         5,0(5)  LD CVTTCBP-PTR TO PTR TO NEXT TCB
000040 4155 0004             00004   39          LA        5,4(5)  BUMP TO ADDR OF PTR TO CURRENT TCB
000044 5855 0000             00000   40          L         5,0(5)  LD PTR TO CURRENT TCB
000048 5865 0018             00018   41          L         6,24(5)  LD TCBMSS
00004C 5876 0008             00008   42          L         7,8(6)  LD HI CORE ADDR
000050 5B76 0004             00004   43 (Exhibit 2) S      7,4(6)  SUBTRACT LO CORE ADDR
```

```
000054 5070 A0EA             000FC    44           ST        7,COREUSED  STORE PARTITION SIZE
000058 4120 A0F6             00108    45           LA        2,SHOW
00005C 5876 0000             00000    46           L         7,0(6)  LD FQE ADDR
000060 5897 0004             00004    47 AGAIN     L         9,4(7)  LD NUMBER OF FREE BYTES
000064 5072 0000             00000    48           ST        7,0(2)  SAVE PTR FOR ABEND DUMP
000068 5092 0004             00004    49           ST        9,4(2)  SAVE PTR FOR ABEND DUMP
00006C 4122 0008             00008    50           LA        2,8(2)
000070 5880 A0EA             000FC    51           L         8,COREUSED
000074 1889                           52           SR        8,9  SUBTRACT FREE AREA FROM PARTITION SIZE
000076 5080 A0EA             000FC    53           ST        8,COREUSED
00007A 5877 0000             00000    54           L         7,0(7)  LD FIRST WORD OF FQE
00007E 5970 A176             00188    55           C         7,=F'0'  AT LAST FQE ?
000082 4770 A04E             00060    56           BNE       AGAIN  NO
  LOC  OBJECT CODE     ADDR1 ADDR2  STMT   SOURCE STATEMENT

000086 5880 A0EA             000FC    57           L         8,COREUSED  YES
00008A 4E80 A0DE             000F0    58           CVD       8,TEMP1
00008E F3F7 A0B3 A0DE 000C5  000F0    59           UNPK      TEMP2,TEMP1
                                      60 *
000094 4180 A0B3             000C5    61           LA        8,TEMP2  EDIT OUT BLANKS
000098 96F0 A0C2       000D4         62           OI        TEMP2+15,X'F0'   FIX ZONE
00009C 95F0 8000       00000         63 GO3       CLI       0(8),C'0'
0000A0 4770 A09E             000B0    64           BNE       GO2
0000A4 9240 8000       00000         65           MVI       0(8),C' '
0000A8 4188 0001             00001    66           LA        8,1(8)
0000AC 47F0 A08A             0009C    67           B         GO3
                                      68 *
0000B0                                69           CNOP      0,4
0000B0                                70 GO2       EQU       *
0000B0 4510 A0C8             000DA    71           BAL       1,WTO
0000B4 0025                           72           DC        AL2(MSGEND-*)
0000B6 0000                           73           DC        AL2(0)
0000B8 5C5C5C40C3D6D9C5               74           DC        C'*** CORE USED'
0000C5 F040404040404040               75 TEMP2     DC        CL16'0'
0000D5 405C5C5C                       76           DC        C' ***'
0000D9                                77 MSGEND    EQU       *
0000DA                                78 WTO       DS        0H
0000DA 0A23                           79           SVC       35
                                      80 *
0000DC 5890 A0E6             000F8    81           L         9,HOLD
                                      82           SPIE      MF=(E,(9))  RESTORE PICA OF CALLING PROG
0000E0 1819                           83+          LR     1,(9) LOAD PARAMETER REG 1
0000E2 0A0E                           84+          SVC    14 ISSUE SPIE SVC
                                      85 *
0000E4 58F0 A0F2             00104    86           L         15,ADDR3
                                      87 *         RELEASE SAVE AREA AND RETURN TO CALLING PROGRAM
```

Exhibit 2 (Continued)

```
0000E8 07FF                       88               BCR         15,15
0000EA 000000000000
0000F0 0000000000000000           89  TEMP1        DC          D'0'
0000F8 00000000                   90  HOLD         DC          F'0'
0000FC 00000000                   91  COREUSED     DC          F'0'
000100 00000000                   92  ADDR1        DC          V(IHESADA)
000104 00000000                   93  ADDR3        DC          V(IHESAFA)
000108 0000000000000000           94  SHOW         DC          32F'0'   SAVE AREA FOR DUMPS
000188                            95               LTORG
000188 00000000                   96                        =F'0'
000000                            97               END         STARTGR
```

Exhibit 2 (Continued)

OS/360 PL/I LINE SPACING

When using List directed I/O in PL/I a blank character is appended to each item to separate each item from the next.

If the width of the item is equal to linesize, then this blank character will appear as the next line. A following SKIP (default to(1)) will cause a space of one line and the blank to be output, resulting in two actual space lines. Example:

```
DCL        X  CHAR (120) INIT ('XYZ');
PUT   LIST (X)    SKIP;
PUT   LIST (X)    SKIP;
```

The output obtained is:

```
XYZ
blank line
blank line
XYZ
```

DOS/360 COBOL ISFMS DATA ON 2321 INDICES ON 2311

The following is a field contribution which has not been submitted to any formal IBM test. Potential users are expected to evaluate its usefulness in their own environment.

The following is a description of a COBOL programming technique used to create a master index for an index sequential file, and to locate the master and cylinder indices on a 2311 disk drive while the prime data is located on a 2321 data cell drive. Both the DOS job control statements and the COBOL routine to alter the skeleton COBOL DTF needed to create these indices are described. The technique was used with Release 13 of DOS/360.

When using COBOL to create an index sequential file the DTF must be altered in order to change the assumed number of overflow tracks, to locate indices on a 2311 when the prime data is on a 2321, or to develop a master index.

To alter the number of cylinder overflow tracks (COBOL assumes 20% of the cylinder for overflow, or 2 tracks for the 2311 and 4 tracks for the 2321) move a hexadecimal representation of the quantity of desired cylinder overflow tracks into byte 21 of the skeleton DTF before the file is opened.

To change the location of the indices to a 2311 when the prime data is on the 2321, move an X'F0' into byte 20 of the skeleton DTF before the file is opened.

To build a master index, the DTF has to be altered both before and after the file is opened.

Move an X'01' into byte 200 of the skeleton DTF before the opening of the file to allow for a master index.

After the open and before the close of the file move an X'02' into byte 65 of the expanded DTF to allow for two levels of indices.

Exhibit 1 shows an example of the calling program. Exhibit 2 shows the called program that allows COBOL to perform the functions stated above.

DOS/360 COBOL ISFMS DATA ON 2321 INDICES ON 2311

```
// JOB SAMPLE    LOAD SAMPLE INDEX SEQUENTIAL FILE        CALLING PROGRAM
// OPTION LISTX,SYM,LOG,LINK,DUMP
  PHASE SAMPLE,*
// EXEC COBOL
001     IDENTIFICATION DIVISION.
001     PROGRAM-ID.      'SAMPLE'
001     ENVIRONMENT DIVISION.
001     CONFIGURATION SECTION.
001     SOURCE-COMPUTER.                IBM-360 F30.
001     OBJECT-COMPUTER.                IBM-360 F30.
001     INPUT-OUTPUT SECTION.
001     FILE-CONTROL.
001         SELECT I-TAPE
001         ASSIGN TO 'SYS021' UTILITY 2400 UNITS.
001         SELECT O-CELL
001         ASSIGN TO 'SYS013' DIRECT-ACCESS 2321
001         RESERVE NO ALTERNATE AREA
001         ORGANIZATION IS INDEXED
001         ACCESS IS SEQUENTIAL
001         RECORD KEY IS I-KEY.

001     DATA DIVISION.
001     FILE SECTION.
                .
                .
                .
001     PROCEDURE DIVISION.
001     START.
001         ENTER LINKAGE.
001         CALL 'CHNGDTFA' USING O-CELL.
001         ENTER COBOL.
001         OPEN INPUT I-TAPE OUTPUT O-CELL.
001         ENTER LINKAGE.
001         CALL 'CHNGDTFB' USING O-CELL.
001         ENTER COBOL.
001     READ-TAPE.
                .
                .
                .
001         CLOSE I-TAPE O-CELL.
001         STOP RUN.
/*
 INCLUDE      MODULE TO ALTER THE DTF
                .
                .
                .
/*
// LBLTYP NSD(07)
// EXEC LNKEDT
// ASSGN SYS021,X'181'
// ASSGN SYS009,X'291'
// ASSGN SYS013,X'297'
// VOL SYS013,SYS013
// DLAB 'SAMPLE INDEX SEQUENTIAL FILE                    1009145',
               0001,67354,99365,'              ',ISC
// XTENT 004,000,000011000,000011009,'009145',SYS009
// XTENT 004,001,000012000,000036009,'009145',SYS009
// XTENT 001,002,000000100,019005419,'011589',SYS013
// XTENT 001,003,100000100,119005419,'011590',SYS013
// XTENT 001,004,200000100,219005419,'011591',SYS013
// XTENT 001,005,300000100,305004419,'011592',SYS013
// XTENT 002,006,305005000,307002419,'011592',SYS013
// PAUSE  MOUNT CORRECT TAPE ON '181'
// EXEC
/*
/&
```

Exhibit 1

DOS/360 COBOL ISFMS DATA ON 2321 INDICES ON 2311

```
// JOB ALTRDTFF   THIS IS A SOURCE LISTING OF THE CALLING PROGRAM
// OPTION LISTX,DUMP,LOG,SYM,LIST,DECK
// EXEC COBOL
002     IDENTIFICATION DIVISION.
002     PROGRAM-ID.      'ALTRDTFF'.
002     ENVIRONMENT DIVISION.
002     CONFIGURATION SECTION.
002     SOURCE-COMPUTER.                        IBM-360 F30.
002     OBJECT-COMPUTER.                        IBM-360 F30.
002     DATA DIVISION.
002     LINKAGE SECTION.
002     01  DTF-FOR-SAMPLE-FILE.
002         02  BYTE OCCURS 250 TIMES          PICTURE X.
002     PROCEDURE DIVISION.
002         ENTER LINKAGE.
002         ENTRY 'CHNGDTFA' USING DTF-FOR-SAMPLE-FILE.
002         ENTER COBOL.
            NOTE MOVE A X'01' TO THE 22ND CHATACTER POSITION OF THE
                SKELETON COBOL DTF,
                THIS WILL ALLOW FOR ONLY ONE OVERFLOW TRACK PER CYLINDER.
002         MOVE 'A' TO BYTE (22). NOTE A IS X'01'.
            NOTE MOVE A X'F0' TO THE 21ST CHARACTER POSITION OF THE
                SKELETON COBOL DTF, THIS WILL ALLOW FOR INDICES
                ON THE 2311 WITH PRIME DATA ON THE 2321.
002         MOVE '0' TO BYTE (21). NOTE 0 IS X'F0'.
            NOTE MOVE A X'01' TO THE 201ST CHARACTER POSITION OF THE
                SKELETON COBOL DTF, THIS WILL ALLOW FOR THE MASTER INDEX.
002         MOVE 'A' TO BYTE (201).  NOTE A IS X'01'.
002          GO TO EXIT.
002         ENTER LINKAGE.
002         ENTRY 'CHNGDTFB' USING DTF-FOR-SAMPLE-FILE.
002         ENTER COBOL.
            NOTE MOVE A X'02' TO THE 66TH CHARACTER POSITION OF DTF,
                THIS WILL ALLOW FOR TWO LEVELS OF INDICES.
002         MOVE 'B' TO BYTE (66). NOTE B IS X'02'.
002     EXIT.
002         ENTER LINKAGE.
002         RETURN.
002         ENTER COBOL.
/*
/&
```

Exhibit 2

S/360-20 DISK SORT CODING SHEET CORRECTION

Installation Newsletter issue 67-24, December 1, 1967, contained an article on the above subject. The statements that the block size must be a multiple of 270 are in error.

The block size and record size should be expressed exactly as they appear on the disk file. The sort program will adjust the I/O area to full sectors.

S/360-20 RPG UNDOCUMENTED MESSAGE

A "Note 333" was encountered during the program listing. No diagnostic message explanation was given and no explanation is given in Appendix D of "Disk and Tape Programming Systems RPG" (C24-9001-3). The cause was determined to be that the File Extension Card was out of order.

S/360-20 DPS RPG INCORRECT CYLINDER INDEX

DPS RPG Index Sequential load does not create the cylinder index correctly. The last logical record in each sector is loaded with X'FF' instead of data. As a result of this, chaining to any record not referenced within the range of the first sector of the CI will fail with a 707, 803 or specification error halt. A solution is to duplicate the last logical record in each sector of CI.

NEW TYPE III PROGRAMS *

The following are the abstracts of programs which have been recently made available from the Type III library.

The program, along with its complete abstract, will be incorporated in subsequent issues of the Catalog of Programs.

Programs may be obtained by submitting a properly completed "General Program Request Card" (Form Number 120-1145-1) to the Program Information Department, 40 Saw Mill River Road, Hawthorne, New York, 10532.

These programs and their related documentation are distributed by IBM in the author's original form and have not been subjected to any formal testing.

Any discussion of Type III Programs must emphasize the following points:

1. Type III programs are not part of the IBM product line, as are Programming Systems (Type I) or Application Programs (Type II).
2. Type III programs have not been subjected to any formal product test.
3. Recipients of Type III programs are expected to make the final evaluation as to the usefulness of the programs in their own environment.
4. There is no committed maintenance for Type III programs. However, any changes the author chooses to make will be announced in subsequent issues of the Catalog of Programs.

* NOTE: THE CUSTOMER MUST BE INFORMED THAT THE ABOVE APPLIES TO ANY OF THE FOLLOWING TYPE III ABSTRACTS FURNISHED TO HIM.

S/360 DISK REMOTE OPERATING SYSTEM (DROS). DROS is a 32K foreground program which provides concurrent job input and output transmissions on multiple switched and non-switched lines via 2701 and 2702 transmission control units. DROS automatically picks up jobs from a terminal job queue and schedules them sequentially in the background partition for execution. Terminal input and output is buffered on a 2311 disk file. Printed and punched output from terminal submitted jobs can be received at the terminal or be directed to the computer center based on submission time instructions in the '//JOB' card.

DROS is written in Assembly Lanugage using BTAM and STRAM. DROS will operate on any System/360 (Model 30 or larger) with 64K, decimal arithmetic, storage protection, and an interval timer. Three 2311 disk files are utilized in DROS for SYSRES, DROS job input and output buffering, and system work files and/or user data files. Terminal service is accomplished through the 2701 and 2702 transmission control units with single drop lines.. DROS supports 1050, 1130, and 1978 terminals. Modification to DOS was kept to an absolute minimum. Only three insertions of code in the Job Control portion of the DOS supervisor are required to allow it to work with DROS.

Ordering Procedure: Order File Number 360D-05.1.011.

The program material can be obtained in card form or on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

The requester must indicate whether a 9 track or 7 track DTR is required. If not specified, a 9 track DTR will be provided.

No tape submittal is required. The DTR will be provided by the Library.

S/360 DEBE-DOS UTILITY PROGRAM. The System/360 DOS Debe Utilities is an adaptation of the original BPS Debe Utilities to operate under 16K DOS. In addition to the original card, tape, and printer functions, DOS Debe also provides 2311 Disk to Printer and 2311 VTOC Display functions, in non-DASD file protect systems. The program is loadable from the core image library of DOS, and requires the 1050 console for function selection and symbolic unit assignments.

Ordering Procedure: Order File Number 360D-00.1.013.

The program material can be obtained in card for form or on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

The requester must indicate whether a 9 track or 7 track DTR is required. If not specified, a 9 track DTR will be provided.

No tape submittal is required. The DTR will be provided by the Library.

S/360-20 CARD RPG COMPILER OVERFLOW PATCH. This patch, consisting of four TXT cards to be inserted in the Model 20 Card RPG Compiler deck (360T-RG010, Version 2) causes page overflow during printing of the RPG source cards at compilation time. Compiler operation is unchanged, with the exception that the diagnostic message, "RG005 CCOL. 6 INCORRECT, CARD IS BYPASSED," does not print in recognizable form due to the use of the area normally occupied by this message as a patch area.

Machine Configuration: As required for 360T-RG010.

Ordering Procedure: Order File Number 360D-03.8.009. Distribution will be in card form only.

S/360 MACRO ASSEMBLER FOR S/360-MODEL 44. The macro assembler for S/360/44 is designed to provide macro capabilities for the Model 44 operating system. The macro package is written in assembler language. It is composed of six programs; 1) assembler phase one, 2) assembler phase two, 3) decoder, 4) encoder, 5) macro library routine and 6) library read routine. Assembler phase one is the main routine. It calls and controls all other routines. The macro language is a subset of OS/360 macro language. This package also has library capabilities for system and user macro libraries.

Order Procedure: Order File Number 360D-03.7.014.

The program material can be obtained on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

The requester must indicate whether a 9 track or 7 track DTR is required. If not specified, a 9 track DTR will be provided.

No tape submittal is required. The DTR will be provided by the Library.

APL/1130-A PROGRAMMING LANGUAGE FOR THE IBM 1130. APL is a conversational implementation of the Iverson notation, an extremely concise mathematical notation with simple but rigorous syntax. This concise attribute virtually eliminates the -program-step in the problem-solving chain of problem--algorithm--program--solution. All operators of the notation and editing capabilities, and the capability to save and retrieve work spaces are provided. The implementation allows data to be structured as scalars, vectors, and matrices with up to 225 elements in any dimension. Numerical values are accurate to six decimal digits, and identifiers are up to 6 alphabetic characters. The program is independent of the IBM monitor, and requires a dedicated disk cartridge. Utilities are provided to generate the system, assign and delete workspaces, and dump/restore individual workspaces and their functions to cards. Minimum configuration is a 1131-2B and a 1442. A 2741 terminal and requisite RPQ are highly desirable.

Ordering Procedure: Order File Number 1130-03.3.001.

Distribution will be in card form only.

ROMANCE-ROY'S METHOD FOR ACCURATE NUMERICAL CONTROL ON THE ELEVEN-THIRTY. The ROMANCE program is a Numerical Control processor designed to facilitate part programming by providing the user with simple methods for tool and pattern definition. Point to Point and simple milling operations can be part programmed through ROMANCE. The programmer has point, line, matrix or circle definitions available, and can invert, repeat, translate, rotate or delete from these patterns. Milling operations which are parallel to the X and Y axis can also be defined. Special features include a data check, the facility of calling pre-defined successions of tools, a guard routine which prevents tool collisions with the work-piece, a sort of output records, and a plot of the tool path.

Minimum configuration - 8K 1130 C. P. U., DISK and either a 1442 CARD READ PUNCH or the 1055 and 1134 TAPE units (an 1132 PRINTER and a 1627 PLOTTER are optional). The program was written in 1130 Fortran IV and runs under the 1130 Monitor System.

Ordering Procedure: Order File Number 1130-23.4.002.

Distribution will be in card form only.

APBE ANALYSIS OF PLACKETT-BURMAN EXPERIMENT. This program computes the mean effects of all independent variables in two level Plackett-Burman experiments where the number of trials is 8, 12, 16, 20, 24, 32, 36, 40, 44, 48, 56, 60, 64, 68, 72, 80, 84, 88, 96. Where the number of variables and trials permit, the program also computes an estimate of the variance.

Programming language is FORTRAN IID.

Machine configuration is: 1620, Model II with 60K memory with auto divide; additional instructions and indirect addressing; 1622 Card Reader Punch; 1443 printer.

Ordering Procedure: Order File Number 1620-06.0.252.

Distribution will be in card form only.

IBM Installation Newsletter

March 22, 1968 Issue No. 68-05

CONTENTS

(G means Green Section)

*Requires Immediate Attention

For IBM Internal Use Only

DPD HQ, White Plains, N.Y.

SELECTED SHORT SUBJECTS

The purpose of Selected Short Subjects is to bring together and highlight concise, factual and timely information which will indicate that action is to be taken by the IBM representative whose accounts are affected.

1. Using the IBM Executary To Improve System Operations

The IBM Executary, when placed at the console of the system, provides a quick and effective method for recording the conditions existing at the time of the occurrence of any incident which affects processing. It provides a history of events and much improved communications between the customer and IBM in identifying and solving machine, programming systems, operator, customer programming, and related problems. See Installation Newsletter issue 68-01, page 18 (green section), for more detail. For information about the IBM Executary, contact your IBM Office Products representative.

OS/360 CUSTOMIZED DISTRIBUTION - PLANNING INFORMATION

The customized distribution service (Letter P67-1) was offered initially with OS Release 9 to allow users having only two 2311 Disk drives to perform system generations more conveniently. The need for this capability was the result of OS/360 library growth.

Since then, the number of components and the size of many existing components has grown, limiting the component selection available for 2-drive users. Further OS growth is expected, and may exceed three 2316 Disk packs in late 1968. As a result of these factors, a plan is being investigated which permits the 2-drive user to generate a greater number of compoments and would reduce the time necessary for PID to ship orders for 2-drive users.

The plan, generally, is to prepare a standard subset package having most of the present OS components (eliminating those which are large-system oriented) packaged on three, rather than two volumes. Users with two drives would be given special system generation instructions and would be able to select any combination of the components in the subset package.

If the outcome of this investigation is positive, it is likely that orders for 2-drive users will require submission of three volumes to PID, possibly as early as Release 15, and that individual customizations will be discontinued.

An additional plan is being investigated with the goal of repackaging the full OS distribution so that all users with three or more Disk drives will be able to use it without needing customization, or subset packaging, after SYS1. MODLIB grows to exceed one 2311 volume capacity.

OS/360 PCP OR MFT CHECKPOINT APPROACH

The following contribution has not been submitted to any formal IBM test. Potential users should first evaluate its usefulness in their own environment.

The following is an approach to avoiding the need to re-enter nucleus initialization parameters after IPLing. A checkpoint of the system is taken after the operator has entered all the OS initialization parameters he intends to run with for the day. A dump of core to tape is taken at this point and then operations proceed as normal. In the event of a system failure, the tape is IPL'ed and operations proceed as normal. The operator has only to reset the clock and resume production. In this manner, the system is restarted in a matter of seconds in the event of an OS/360 system failure.

A flow chart of the approach is shown in Exhibit 1.

1. Modification to OS/360

   SVC IG C0203D-Modified to recognize an additional display command parameter 'DISPLAY DUMP'. The command will XCTL to SVC IGC0303D.

2. Additions to OS/360

a) SVC IGC0303D

1. ATTRIBUTES-REUSABLE, TRANSIENT

2. FUNCTION-DUMP the contents of core to a predetermined 9-track tape drive in a form that is 'IPL-ABLE'.

3. ENTRIES-The only entry to this module is from IGC0203D VIA an XCTL.

4. MACROS-Macros used in this module

a. GET MAIN
b. ABEND
c. FREEMAIN
d. EXIT

5. INPUT-At entry to this module, register 6 points to 40 bytes of core that is used to pass information from MEXCP to IEC0103D. These 40 bytes carry the following information:

Bytes 00-15 calling sequence used to XCTL to

Bytes 16-23 verb from operator command

Bytes 24-39 Parameter list from operator command.

3. Description of module ICC0303D

The program determines how much core is available by looking at the CVT. From this information a list of CCW's are constructed to both write out and read in core. The OS IO interrupt new PSW is changed to point to the IO interrupt handler in this module and is later restored to its original value. The program starts IO to a predetermined 9-track tape unit, then ends via SVC EXIT.

When the checkpoint tape is IPL'ed, control is transferred to the restore segment of the program which continues reading in the checkpoint tape and then ends via SVC EXIT.

4. Restriction

TIMER MUST BE UPDATED with set clock command after IPLing CHECKPOINT tape.

5. Advantage

The time it takes to restart a 512K system after a system failure is approximately six seconds.

OS/360 FORTRAN H VERSION II CHANGE IN PASSING NAMES

The method used by FORTRAN H to pass subroutine or function names in a call statement has been changed with FORTRAN H Version II. Names of functions and subroutines being passed as arguments in a CALL statement must appear in an external statement in the calling routine. In routines compiled under FORTRAN H Version I, the calling routine would pass an ADCON, which was in the parameter list, to the routine being called. This ADCON would point directly to the entry point of the function or subroutine being passed. Routines compiled under Version II, however, pass not the ADCON itself, but a pointer to the ADCON.

There are two instances where some action is required. The first is where subroutines or a function name is being passed between the routine compiled under FORTRAN H Version II and a routine written in Assembler language. If the Assembler language routine was written to operate with FORTRAN H Version I, the interface will have to be changed to accept a pointer to the ADCON instead of the ADCON itself. The second instance is where a subroutine or function name is being passed between a routine compiled under FORTRAN H Version I and one compiled under FORTRAN H Version II.

OS/360 PCP OR MFT CHECKPOINT APPROACH

Flow Chart of SVC IGC0303D

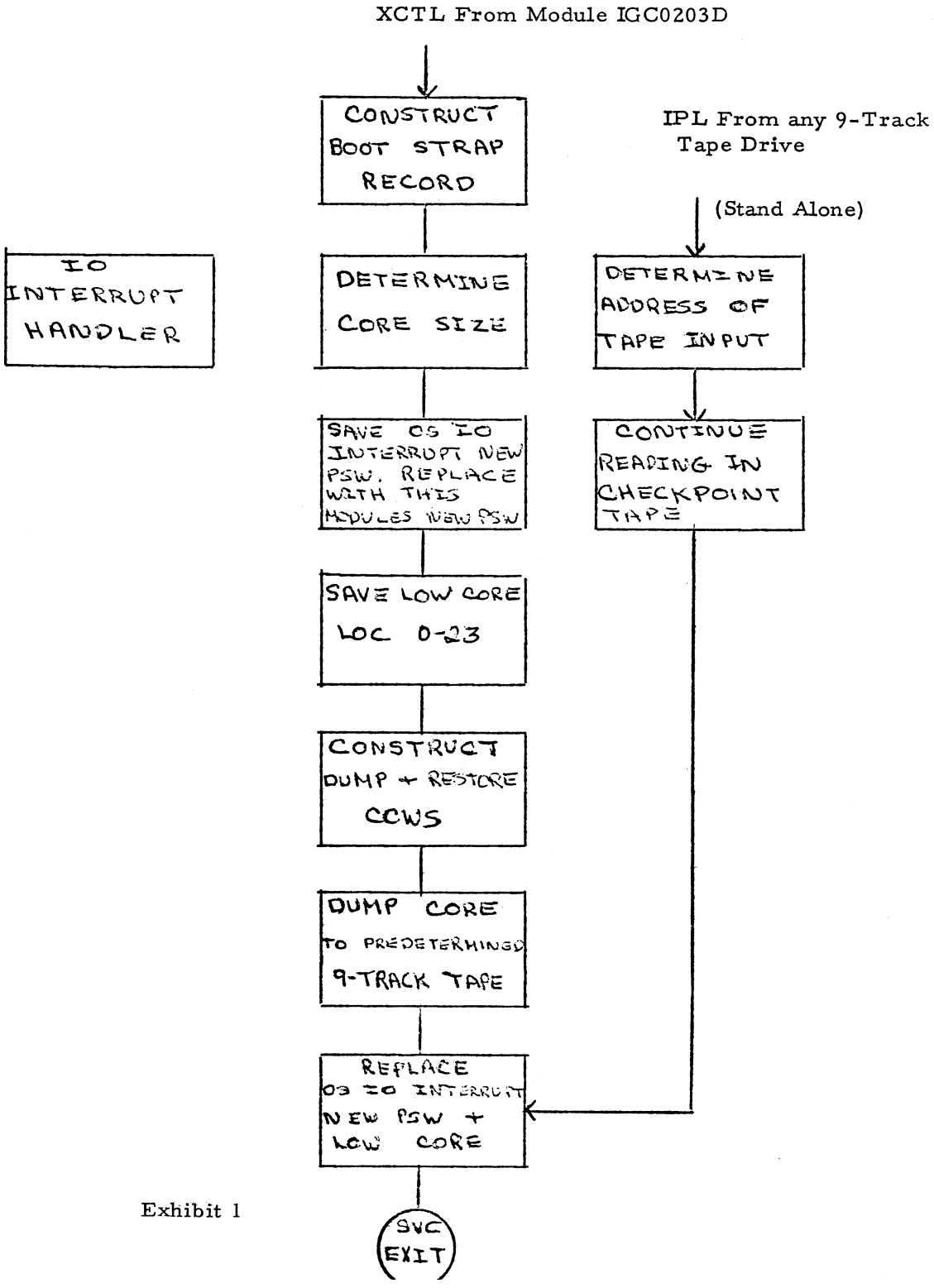


Exhibit 1

For IBM Internal Use Only

In order to operate properly, a routine compiled under Version I will have to be recompiled under Version II.

The reason for this change was to make all OS FORTRANs compatible. Difficulties similar to those discussed above had previously existed when passing a subroutine or function name between the routine compiled under FORTRAN H and another compiled under either FORTRAN E or FORTRAN G. This inconsistency now has been eliminated.

OS/360 MINIMUM CORE REQUIREMENT FOR 2314 SORT

There is no explicitly stated minimum core storage for the 2314 sort in the SRL Manuals. The actual minimum is variable, and averages 25K. The minimum should be considered to be 44K to obtain good performance. It is presently planned to include a more explicit statement concerning the 2314 sort main storage requirements in a future SRL revision.

S/360-20 NEW MODELS-DISK SORT/MERGE TIMING

The following information is preliminary, subject to change, and is provided for planning purposes only. Actual timings and specifications will be provided through the official IBM documentation procedures.

The timings apply to the newly announced models of S/360-20. See Product Announcement letter 268-21, dated 3/5/68 and the Sales Manual pages attached thereto.

1. Performance:

The sort time estimates in the table below are based on the following assumptions:

a) Sort/Merge program is disk-resident.

b) One control field.

c) Disk input and output.

d) No options used.

e) Work areas on different drives for two-drive sorts.

f) Freeout specified.

Time in Minutes

Record Length	File Size	12 K Core		16 K Core	
		1 Drive	2 Drives	1 Drive	2 Drives
80	5000	7	5	6	5
	10000		9		9
100	5000	7	6	6	5
	10000		12		9
200	5000		9		8
270	4200		15		13

2. Minimum System Requirements

A 2020 Processing Unit Model BC4 (12K), a 2560 Model A2, a 2311 Model 12 Disk Storage Drive, and if error and diagnostic messages are desired, a 2203 Model A2 Printer.

S/360-20 TPS/DPS - RETURNING CONTROL TO JOB CONTROL PROGRAM

The following is a field contribution which has not been submitted to any formal IBM test. Users should evaluate its usefulness in their own environment prior to implementation.

When operating under S/360-20 TPS or DPS, it is sometimes desirable to return control to the Job Control Program in the event of an error stop (non-programmed halt) - in order to initiate the execution of the next program in the job stream.

Control can be returned to the Job Control Program at anytime by altering the contents of Register 3 (actually the instruction address portion of the PSW) to 00C2 and pressing start on the CPU console. This procedure eliminates the need to re-IPL in such a situation.

S/360-20 DISK RPG CHAINING ERRONEOUS DIAGNOSTIC

In using S/360-20 Disk RPG, there seems to be a problem in generating a program when one file with two chaining fields is used to chain to two indexed sequential files.

An erroneous diagnostic (message 236) is given. It states the chaining field and the key field to the second chained file have unequal lengths.

Contact the servicing Field Engineering Branch Office regarding a PTF to correct this problem.

1130 IMPLEMENTING MONITOR SYSTEM VERSION II

Version II of the 1130 monitor, supporting the new features and models was released on December 15, 1967. Some improvements in the system are:

1. Data Statement in FORTRAN
2. Dynamic Dump
3. Faster Loading of Core Image Programs
4. Dump Data Directly From Users Area
5. Extended Assembler Mnemonics

Some customer reprogramming may be required for operation of existing programs under Version II.

It is important that prior to working with Version II, customers obtain and review the following SRL's for instructions:

C26-3709-0 IBM 1130 Disk Monitor System Version II, System Introduction

C26-3717-0 IBM 1130 Disk Monitor System Version II Programming and Operators Guide.

INFORMATION ABOUT THE NEWSLETTER

EVERY SALESMAN AND SYSTEMS ENGINEER SHOULD RECEIVE ONE COPY OF THE NEWSLETTER. Each FE Branch Manager should receive five copies for distribution to Customer Engineers.

The IBM Installation Newsletter is distributed only to IBM locations and is not available to customers. The Newsletter is separated into two parts. The first part contains information which is intended only for IBM personnel and the pages are labeled For IBM Internal Use Only. THESE PAGES, OR REPRODUCTIONS OF THEM, ARE NOT TO BE GIVEN TO CUSTOMERS. However, using your own good judgement, you may discuss pertinent information from these pages with a customer. The second part contains information which may be REPRODUCED by the Branch Office at their discretion and given to customers. These pages are not labeled with any restrictive classification. Note that when any abstracts for Type III programs are given to the customer, he must be informed that the disclaimer for Type III programs as contained in the Newsletter, applies. YOUR COOPERATION IN PROPERLY HANDLING THE CONTENT OF THE IBM INSTALLATION NEWSLETTER IS NECESSARY FOR ITS CONTINUED PUBLICATION.

Changes to the quantity of the Newsletter shipped are to be made by the Administration Manager. (NOTE THAT EXTRA COPIES OF THE NEWSLETTER ARE NOT TO BE ORDERED

FOR DISTRIBUTION OF INFORMATION TO CUSTOMERS.) He should update and send the DP Administration Managers' Communication Control card (form number N44-780) to DP Communication Control, DPD HQ.

Direct inquiries concerning the contents of this Newsletter to:

W. P. Gaffigan, Editor
IBM Installation Newsletter
Installation Planning-#713
DPD HQ
112 East Post Road
White Plains, New York 10601
Telephone 914 WH9-1900 Extension 4781

NEW PUBLICATIONS ANNOUNCED BY PRL'S

The weekly PRL's (Publications Release Letters) are used to insure that all Salesmen and Systems Engineers are aware of new or revised Marketing Publications. Normally, each issue of the Newsletter will contain information from two PRL's, one following the other. The information will be placed in the Newsletter in its original form with no rearrangement of form numbers or titles. It is not intended to replace existing information and distribution sources. You should be certain that you are aware of these sources.

Distribution: Branch Offices-DP Management, Salesmen, Systems Engineers, FE Managers, Regions, Districts, Education Centers, Field Systems Centers, Federal Systems Centers, FE Area Offices, DPD HQ, FED HQ, WTC.

PRL #9 March 1, 1968

REFERENCE SOURCES

FORM NO.	TITLE	DISPOSITION
Z20-1873-0	KWIC Index to Tie Re: Z20-0305-16	NEW

MARKETING PUBLICATIONS

221-0369-2	The IBM 2250 Display Unit, Model 1 Proposal Insert	NEW - Reinstated
221-0415-2	The IBM 2260 Display Station Proposal Insert	Use 221-0415-1
221-0481-0	The IBM 2250 Display Unit Model 3 Proposal Insert	NEW
322-0037-0	IBM Journal of Research and Development, Volume 11, No. 6	NEW
322-0038-0	IBM Journal of Research and Development, Volume 12, Number 1	NEW
520-2065-1	IBM System/360 Model 85 Facts Folder	Scrap 520-2065-0
A21-9031-1	IBM 1231-N1 Component Description and Operating Procedures	Use A21-9031-0, N21-0089, N21-0061 and N21-0068
A22-6823-05	IBM System/360 Input/Output Configurator	Scrap A22-6823-4
A22-6824-4	IBM S/360 Data Communications and Acquisition Configurator	Use A22-6824-3 & N27-3009
A24-3021-4	IBM 1030 Data Collection System Installation Manual - Physical Planning	Use A24-3021-3 & N27-3015
C24-3450-3	IBM S/360 BOS Operating Guide	Scrap C24-3450-2, N24-5132, N24-5161, N24-5260 & N24-5321. Also scrap C24-3452-3 with N21-5071. C24-3453-0 and -1 with N21-5067 and N24-5082. C24-3454-0
E20-8034-1	Inventory Control and Material Accounting	Scrap E20-8034-0
H20-0331-2	System/360 APT Numerical Control Processor (360A-CN-10X) Operator's Manual Version 2	Scrap H20-0331-1
K20-0083-0	Austin National Bank Proof and Transit with IBM 1260 Electronic Inscribers and the IBM S/360	NEW
L27-3008-0	Flight Data Entry and Printout Equipment 1980 Model 1, 1051 Model 2, 1926 Model 15 RPQ	NEW
L27-3009-0	IBM 2905 Remote Multiplexer	NEW
N20-1049	TNL to General Purpose Simulation System/360 Application Description Re: H20-0186-2	NEW
N20-1078	TNL for System/360 Mortgage Loan Program (360A-FB-19X) Re: H20-0469-0	NEW
N20-1848	TNL to 1130 Continuous System Modeling Program (1130-CX-13X) Program Reference Manual Re: H20-0282-0	NEW
N23-0613-0	ENL to IBM S/360 DOS/ TOS Coding Guide Re: R20-1039-1	NEW
N27-1290	IBM S/360 Operating System Planning for Multi-Programming with a Fixed Number of Tasks Version II (MFT II) Re: C27-6939-0	NEW
N27-2921	System/360 Model 65 Operating Procedures Re: A27-2728	NEW
N27-3021	Changes to 1050 Data Communications System Principles of Operation Manual Re: A24-3474-0 with N27-3013	NEW
N27-3022	TNL 2780 Data Transmission Terminal Physical Planning Re: A27-3007	NEW
N30-2509	TNL-IBM S/360 Operating System Remote Job Entry Re: C30-2006-0 with N30-2506	NEW
N33-8517	TNL to Mod. 20 IOCS Punched Card Equipment Operating Procedures Re: C26-3803-0, -1 with N24-9013	NEW
R20-1019-4	IBM S/360 PL/I Coding Practice Problems	Use R20-1019-3
R20-1040-1	IBM S/360 PL/I Coding (DOS/TOS Subset) Supplementary Education Guide	Use R20-1040-0
R20-4075-0	IBM S/360 General Purpose System Simulator Education Guide	NEW
R29-0231-3	IBM S/360 ALC P.I. Systems Review Text	Use R29-0231-2
R29-0256-0	IBM System/360 Introduction P.I. Text	NEW
NFN	Photographs (IBM Products - IBM Buildings)	Revised Abstract
V20-6061-0	IBM System/360 Model 85 Flip Chart	NEW
Y24-5068-0	TNL to IBM S/360 DOS/TOS Ut. Macros PLM 2314 Support Re: Y24-5045-1	NEW
Y28-2291-0	IBM S/360 Disk Operating System: Sort/Merge, PLM Re: Y24-5021-0	NEW
Y33-8506-0	TNL to S/360 Model 20 TPS Utility Programs PLM Re: Y33-9006-0	NEW
Y33-8507-0	System/360 Model 20 TPS Sort/Merge PLM Re: Y33-9005-0	NEW
Y33-9000-1	IBM S/360 Mod 20 Tape Programming Sys;tem Control and Service Programs PLM	Scrap Y33-9000-0
Z77-8013-0	Reconstruction of a Nicked or Damaged Strip in the Middle of an Index Sequential File on the Data Cell	NEW
Z77-8015-0	Overlaying DOS Sorts with COBOL Subroutines and Executing in the Foreground area	NEW
Z77-8017-0	Multiprogramming an IBM 1418 Optical Character Reader Under DPS	NEW

KWIC INDEX FOR NEWSLETTER ISSUES 67-05 THROUGH 68-04

For IBM Internal Use Only

For IBM Internal Use Only

For IBM Internal Use Only

IBM INSTALLATION NEWSLETTER

TABLE OF CONTENTS

OS/360 SYSTEM GENERATION CHANGES FOR MFT II

To make the conversion from MFT I to MFT II, the following changes are required to the SYS-GEN deck:

1. Macros involved:

   CTRLPROG- changes
   SUPRVSOR - changes
   SCHEDULR - changes
   PARTITNS - new

2. Changes:

   a. In CTRLPROG, HITASK and LOWTASK must be deleted. This function has been taken over by the PARTITNS macro. There is also an additional parameter (SYSQUE=size), which has an optional default of 16ØØ.

   b. In SUPRVSOR, WAIT=SINGLE is now invalid for MFT II.

   c. In SCHEDULR, TYPE=SEQUENTIAL must be changed to TYPE=MFT. The following parameters are no longer valid for MFT: TSYSIN, TSYSOUT RESJOBQ, CANCEL, VLMOUNT, TAVR. The DESIGN parameter for MFT II will only support 26K or 44K. Two new parameters have been added for MFT II: (WTOBFRS and REPLY). They are both optional. If omitted, a value of 5 is assumed.

   d. See the PARTITNS macro to define partition sizes.

OS/360 USER REFERENCES TO SYSTEM CONTROL BLOCKS

The SRL publication IBM System/360 Operating System-System Control Blocks (C28-6628) contains the details of many system "internal" formats. Users who refer to these control blocks in any way other than by normal system macro-instructions become dependent on a specific OS release, since the internal control block formats are subject to change as part of normal development and maintenance. Programs which depend on specific control block formats should be written and used with the clear understanding that such programs may need revision with each OS release.

OS/360 SPACE ALLOCATION FOR A BDAM DATA SET

Installation Newsletter issue 67-21 contained an article with the above title, on page 3. Some confusion exists and the following changes should be made to that article.

1. The first sentence should be changed to read: "When creating a multi-volume BDAM Data Set, Space Allocation is accomplished by use of primary and secondary allocation".

2. The last sentence should be changed to read: "The above applies to all OS/360 BDAM".

OS/360 2540 STACKER SELECT ROUTINE

The following is a field contribution which has not been submitted to any formal IBM test. No maintenance can be expected. Potential users should first evaluate its usefulness in their own environment.

This routine (UTAU13) is in use in a customer installation. It can be called by a COBOL or Assembly Language program to select cards into the appropriate stacker, including stacker 3. The routine is shown in Exhibit 1.

OS/360 2540 STACKER SELECT ROUTINE

PAGE

```
STMT   SOURCE STATEMENT                                               F01JUN67   1/25/6

  2 ************************************************************************ 0C00010
  3 **** UTAU13 MAY BE CALLED BY ANY OPERATING SYSTEM PROGRAM. WITH VERY ** 0C00010
  4 **** LITTLE MODIFICATION ANY SYSTEM 360 PROGRAM MAY USE IT.          ** 0000010
  5 **** THIS IS EXECUTE CHANNEL PROGRAM THAT OPENS,READS,STACKER SELECTS** 0C00011
  6 **** AND CLOSES A CARD DATA SET. IT HAS A DDNAME OF SYSIN. TWO PARAME** 0C00012
  7 **** TERS ARE PASSED FOR OPEN, CLOSE, AND STACKER SELECT. THREE      ** 0000013
  8 **** PARAMETERS ARE PASSED FOR THE READ. A STATUS CODE IS ALWAYS     ** 0000013
  9 **** RETURNED TO THE USER.                                           ** 0000014
 10 **** THIS PROGRAM MAY BE CALLED BY A COBOL OR ALP PROGRAM. THE       ** CC00014
 11 **** CALLING PROGRAM SHOULD ALWAYS USE THE NAME OF UTAU13.           ** 0C00014
 12 ************************************************************************ 0C00015
 13 *   REG. 6 HAS FIRST PARAMETER PASSED                                    0000020
 14 *   REG. 7 HAS STATUS ADDRESS PASSED AND COMPLETION CODE PASSED          0C00030
 15 *   REG. 8 HAS CARD READ AREA ADDRESS                                    C000040
 16 *   COMP. CODE A IS SUCCESSFUL COMPLETION OF ANY FUNCTION                CC00050
 17 *   COMP. CODE B IS INVALID OPEN                                         CC00060
 18 *   COMP. CODE C IS INVALID OPERATION CODE BEING PASSED-NOT 1,2,3,4,5,6  CC00070
 19 *   COMP. CODE D IS A READ OF UNOPENED DATA SET                          0000080
 20 *   COMP. CODE E IS WHEN '7F' OR '41' IS NOT IN ECB ON READ              0000090
 21 *   COMP. CODE F IS WHEN '01' OR '02' IS NOT STATUS IN CSW ON READ       0000100
 22 *   COMP. CODE G IS A PERM. I-O ERROR - RERUN JOB  - READ                0C00110
 23 *   COMP. CODE H IS BUS-OUT ERROR - RERUN JOB     - READ                 CC00120
 24 *   COMP. CODE I IS EQUIP.OR DATA CHECK - CHECK CARDS AND RERUN JOB-READ 0000130
 25 *   COMP. CODE J IS END OF DATA                                          0C00140
 26 *   COMP. CODE K IS WHEN '7F' OR '41' IS NOT IN ECB ON STACKER SELECT    0000150
 27 *   COMP. CODE L IS WHEN '02' IS NOT STATUS IN CSW FOR STACKER SELECT    CC00160
 28 *   COMP. CODE M IS AN ERROR THAT CANNOT HAPPEN-I HOPE-STACKER SELECT    0000170
 29 *   COMP. CODE N IS BUS-OUT ERROR - RERUN JOB     -STACKER SELECT        000018C
 30 *   COMP. CODE P IS INVALID COMMAND - STACKER SELECT                     0000190
 31 *   COMP. CODE Q IS UNSUCCESSFUL CLOSE                                   0C00200
 32 *                                                                        0C00210
 33 UTAU13    START 0                  THIS IS A BEAUTIFUL START             CC00220
 34           USING *,15                                                     0000230
 35           SAVE  (14,12)                                                  0C00240
 36+          CS    CH
 37+          STM   14,12,12(13) SAVE REGISTERS
 38           LM    6,10,0(1)          LOAD COBOL AREAS IN REG.6-10          CC00250
 39           BALR  5,0                                                      00002601
 40           USING *,5                                                      00002701
 41           ST    13,MYSAVE+4                                              00002801
 42           LA    13,MYSAVE                                                00002901
 43 GO        CLI   0(6),C'4'          OPEN ROUTINE                          00003001
 44           BE    OPENA                                                    0C003101
 45           CLI   0(6),C'5'          READ A CARD                           CC003201
 46           BE    READA                                                    0000330(
 47           CLI   0(6),C'1'          STACKER 1                             0C00340(
 48           BE    STACK1                                                   0000350I
 49           CLI   0(6),C'2'          STACKER 2                             CC00360(
 50           BE    STACK2                                                   0000370(
 51           CLI   0(6),C'3'          STACKER 3                             0000380(
 52           BE    STACK3                                                   0000390(
 53           CLI   0(6),C'6'          CLOSE ROUTINE                         CC00400(
 54           BE    CLOSEA             CLOSE ROUTINE                         CC00410(
 55           MVI   C(7),C'C'          INVALID OPERATION CODE                0000420(
 56           B     RET                                                      0000430(
 57 OPENA     OPEN  (CARD,(INPUT))                                           0000440(
```

Exhibit 1

OS/360 2540 STACKER SELECT ROUTINE

```
STMT   SOURCE STATEMENT                                          F01JUN67   1/25/(
  58+          CNOP  0,4
  59+OPENA     BAL   1,*+8 LOAD REG1 W/LIST ADDR.
  60+          DC    AL1(128) OPTION BYTE
  61+          DC    AL3(CARD) DCB ADDRESS
  62+          SVC   19 ISSUE OPEN SVC
  63           TM    CARD+48,B'00010000'      CHECK FOR OPEN SUCCESS        0000450
  64           BO    RETRN                                                  0000460
  65           MVI   0(7),C'B'                DID  NOT OPEN                 0000470
  66           B     RET                                                    0000480
  67 READA     TM    CARD+48,B'00010000'      CHECK TO SEE IF OPEN          0000510
  68           BO    A1                                                     0000520
  69           MVI   0(7),C'D'                NOT OPEN                      0000530
  70           B     RET                                                    0000535
  71 A1        MVI   RDCCW,X'C2'              MOVE READ OP CODE TO CCW      0000540
  72           ST    8,S8                     STORE ADDRESS OF IO AREA      0000550
  73           MVC   RDCCW+1(3),S8+1          MOVE ADDRESS OF CARD AREA     0000560
  74           NI    CDECB,X'00'              RESET ECB FLAG BYTE           0000570
  75           EXCP  CDIOB                                                  0000580
  76+          LA    1,CDIOB LOAD PARAMETER REG 1
  77+          SVC   0 ISSUE SVC FOR EXCP
  78           WAIT  ECB=CDECB                                              0000590
  79+          LA    1,CDECB LOAD PARAMETER REG 1
  80+          LA    0,1(0,0) COUNT OMITTED,1 USED
  81+          SVC   1 LINK TO WAIT ROUTINE
  82           TM    CDECB,X'7F'              SUCCESSFUL COMPLETION         0000600
  83           BO    A2                       GOOD                          0000610
  84           TM    CDECB,X'41'              ERROR                         0000620
  85           BO    A3                       FIND OUT PROBLEM              0000630
  86           MVI   0(7),C'E'                MOVE IN ERROR CODE            0000650
  87           B     RET                                                    0000660
  88 RETRN     MVI   0(7),C'A'                MOVE IN GOOD CODE             0000670
  89           B     RET                                                    0000680
  90 A2        CLC   0(2,8),=C'/*'            SLASH *                       0000690
  91           BE    END                                                    0000700
  92           B     RETRN                                                  0000710
  93 A3        TM    CSWCD+3,X'01'            UNIT EXCEPTION - END OF DEVICE 000720
  94           BO    END                                                    000730
  95           TM    CSWCD+3,X'02'            UNIT CHECK                    0000740
  96           BO    A4                                                     0000750
  97           MVI   0(7),C'F'                MUST BE ONE OF ABOVE          0000760
  98           B     RET                                                    0000770
  99 A4        TM    SENSECD,X'40'            INTERVENTION REQUIRED         000780
 100           BO    B4                                                     0000790
 101           TM    SENSECD,X'20'            BUS-OUT                       0000800
 102           BO    A6                                                     000810
 103           TM    CSWCD+3,X'0E'            BIG PROBLEM                   000820
 104           BO    A7                                                     0000830
 105           MVI   0(7),C'G'                                              000840
 106           B     RET                                                    000850
 107 A6        MVI   0(7),C'H'                BUS-OUT                       0000880
 108           B     RET                                                    000890
 109 A7        MVI   0(7),C'I'                EQIP OR DATA CHECK - RERUN    000900
 110           B     RET                                                    000910
 111 END       MVI   0(7),C'J'                END OF DATA                   000920
 112           MVI   EOJSW,X'AA'              SET SWITCH  TO RETURN FROM SS 000920
 113           B     STACK1                   STACK THE /* CARD IN  POCKET 1 0000930
```

Exhibit 1 (Continued)

OS/360 2540 STACKER SELECT ROUTINE

```
STMT   SOURCE STATEMENT                                        F01JUN67   1/25/68

114 RET      L     13,MYSAVE+4          ************************************ 00009400
115          RETURN (14,12)                                                00009500
116+         LM    14,12,12(13) RESTORE THE REGISTERS
117+         BR    14 RETURN
118 ****  THIS ROUTINE FOR STACKER SELECT 1  *********                      00009700
119 STACK1   MVI   RDCCW,X'23'          STACKER SELECT POCKET 1             00009800
120 STACK    NI    CDECB,X'00'                                             00009900
121          EXCP  CDIOB                                                   00010000
122+         LA    1,CDICB LOAD PARAMETER REG 1
123+         SVC   0 ISSUE SVC FOR EXCP
124          WAIT  ECB=CDECB                                               00010100
125+         LA    1,CDECB LOAD PARAMETER REG 1
126+         LA    0,1(0,0) COUNT OMITTED,1 USED
127+         SVC   1 LINK TO WAIT ROUTINE
128          CLI   ECBSW,X'AA'          SEE  IF  THIS IS  LAST CARD /*     00010140
129          BE    RET                  RETURN TO CALLING PROGRAM          00010160
130          TM    CDECB,X'7F'          SUCCESS                            00010200
131          BO    RETRN                                                   00010300
132          TM    CDECB,X'41'          ERROR                              00010400
133          BO    B2                                                      00010500
134          MVI   0(7),C'K'                                               00010600
135          B     RET                                                     00010700
136 B2       TM    CSWCD+3,X'02'        UNIT CHECK-NOTHING ELSE IS VALID   00010800
137          BO    B3                                                      00010900
138          MVI   0(7),C'L'                                               00011000
139          B     RET                                                     00011100
140 B3       TM    SENSECD,X'40'        INTERVENTION REQUIRED              00011200
141          BO    B4                                                      00011300
142          TM    SENSECD,X'20'        BUS-OUT                            00011400
143          BO    B5                                                      00011500
144          TM    SENSECD,X'80'        COMMAND REJECT                     00011600
145          BO    B6                                                      00011700
146          MVI   0(7),C'M'            ERROR                              00011800
147          B     RET                                                     00011900
148 B4       WTO   'READY READER'                                          00012000
149+         CNOP  0,4
150+B4       BAL   1,IHB0012A BRANCH AROUND MESSAGE
151+         DC    AL2(IHB0012-*) MESSAGE LENGTH
152+         DC    AL2(0)
153+         DC    C'READY READER' MESSAGE
154+IHB0012  EQU   *
155+IHB0012A DS    0H
156+         SVC   35 ISSUE SVC
157          B     GO                   FIND OUT WHICH STACKER             00012100
158 B5       MVI   0(7),C'M'                                               00012200
159          B     RET                                                     00012300
160 B6       MVI   0(7),C'P'            INVALID COMMAND                    00012400
161          B     RET                                                     00012500
162 *******  STACKER SELECT POCKET 2   ************************************ 00012550
163 STACK2   MVI   RDCCW,X'63'          STACKER NO. 2                      00012600
164          B     STACK                                                   00012700
165 *******  STACKER SELECT 3  ********                                     00012800
166 STACK3   MVI   RDCCW,X'A3'          STACKER NO. 3                      00012900
167          B     STACK                                                   00013000
168 ****     CLOSE  ROUTINE  ****                                          00013100
169 CLOSEA   CLOSE (CARD)                                                  00013200
```

Exhibit 1 (Continued)

OS/360 2540 STACKER SELECT ROUTINE

```
PROGRAM-IBM

STMT   SOURCE STATEMENT                                          F01JUN67   1/25/68

170+           CNOP  0,4
171+CLOSEA     BAL   1,*+8 BRANCH AROUND LIST
172+           DC    AL1(128) OPTION BYTE
173+           DC    AL3(CARD) DCB ADDRESS
174+           SVC   20 ISSUE CLOSE SVC
175            TM    CARD+48,B'00010000' CHECK FOR CLOSE                  00013300
176            BO    C1                                                   00013400
177            B     RETRN                                                00013500
178 C1         MVI   0(7),C'Q'           UNSUCCESSFUL CLOSE               00013600
179            B     RET                                                  00013700
180 ****   CARD IOB   ********************************************************* 00015000
181            DS    0F                                                   00015100
182 CDIOB      DS    0CL32                                                00015200
183 FLAGCD     DC    X'02'               FLAG                             00015300
184            DC    X'00'                                                00015400
185 SENSECD    DC    H'0'                                                 00015500
186            DC    X'00'                                                00015600
187            DC    AL3(CDECB)          ECB ADDRESS                      00015700
188            DC    X'0'                                                 00015800
189 CSWCD      DC    XL7'0'                                               00015900
190            DC    X'0'                                                 00016000
191            DC    AL3(RDCCW)          CCW ADDRESS                      00016100
192            DC    X'0'                                                 00016200
193            DC    AL3(CARD)           DCB ADDRESS                      00016300
194            DC    XL2'0'                                               00016400
195 BLKCD      DC    XL2'0'                                               00016500
196            DC    XL2'0'                                               00016600
197 **** CCW STRING *** FOR READ                                          00016700
198 RDCCW      CCW   X'02',S8,X'00',80                                    00016800
199 CARD       DCB   DSORG=PS,MACRF=(E),DDNAME=SYSIN,DEVD=RD,BUFNO=1
200                  *,***  IHB063  DDNAME SHORT-PADDED TO 8 CHAR

202+*                          DATA CONTROL BLOCK
203+*
204+           CRG   *-16 TO ELIMINATE UNUSED SPACE
205+CARD       DS    0F ORIGIN ON WORD BOUNDRY
206+           ORG   *+16 TO ORIGIN GENERATION

208+*                          READER/PUNCH DEVICE INTERFACE

210+           DC    BL2'0000000000000000' MODE,STACK,DEVT
211+           DC    H'0'

213+*                          COMMON ACCESS METHOD INTERFACE

215+           DC    AL1(1) BUFNO
216+           DC    AL3(1) BUFCB
217+           DC    AL2(0) BUFL
218+           DC    BL2'0100000000000000' DSORG
219+           DC    A(1) IOBAD

221+*                          FOUNDATION EXTENSION

223+           DC    BL1'00000000' BFTEK,BFALN
224+           DC    AL3(1) EODAD
```

Exhibit 1 (Continued)

'ROGRAM-IBM OS/360 2540 STACKER SELECT ROUTINE

```
STMT  SOURCE STATEMENT                                        F01JUN67   1/25/6

225+          DC    BL1'00000000' RECFM
226+          DC    AL3(0) EXLST

226+*                          FOUNDATION BLOCK

230+          DC    CL8'SYSIN' DDNAME
231+          DC    BL1'00000010' OFLGS
232+          DC    BL1'00000000' IFLG
233+          DC    BL2'1101000000000001' MACR
234 **** CARD ECB ********************************************** 0001700
235 CDECB     DC    F'0'                                          0001710
236 MYSAVE    DC    18F'0'                                        0001720
237 S8        DC    F'0'                                          0001730
238 ECJSW     DC    X'00'              END OF JCB SWITCH          0001740
239           LTORG                                               0008888
240                 =C'/*'
241           END                                                 0009999
```

Exhibit 1 (Continued)

OS/360 - RELEASE 13 DISPLAY TAPES - NOT U/R CORRECTION

The following has been supplied by the author as a correction to the article "OS/360 Release 13-Displaying Tape Units Instead of U/R Devices" which appeared on page 2 of Installation Newsletter issue 68-01.

APAR P12993 affects the use of this modification. SUPERZAP to fix this apar in SYSLIB (DSNAME is SYS1.LINKLIB) follows:

```
NAME IEFCNTRL XTTP00
VERIFY 00FC 954A7709
REP    00FC 954E7709
VERIFY 019E 954A7709
REP 019E 954E7709
VERIFY 01CE 954A7709
REP 01CE 954E7709
VERIFY 0224 924A7709
REP 0224 924E7709
```

OS/DOS/360 COBOL D & E SUBPROGRAM CALL EFFICIENCY

The following is a contribution from an IBM Field Systems Center. It has not been submitted to any formal IBM test. Potential users should first evaluate its usefulness in their own environment.

In order to speed up COBOL program execution and reduce core requirements, a call to a subprogram should attempt to minimize the number of parameters passed. This may be accomplished by passing just one dummy "01" level parameter with the true arguments specified as sub-levels.

NOTE: CARE MUST BE EXERCISED IN ASSURING THAT THE PARAMETERS ARE CONTIGUOUS AND THAT THEY ARE DEFINED IDENTICALLY IN BOTH THE CALLED AND CALLING SUBPROGRAMS SO AS TO TAKE INTO ACCOUNT ALIGNMENT, SLACK BYTES, ETC.

1. A Method

Calling Program	Called Program
Working-Storage Section	Linkage Section
02 A Picture X	02 C Picutre X
02 B Picture X	02 D Picture X
Procedure Divison	Procedure Divison
Enter Linkage	Enter Linkage
Call 'SUB' using A, B	Entry 'SUB' using C, D
Enter COBOL	Enter COBOL

2. Better Method

Calling Program	Called Program
Working-Storage Section	Linkage Section
01 OVALL1	01 OVALL2
02 A Picture X	02 C Picture X
02 B Picture X	02 D Picture X
Procedure Divison	Procedure Divison
Enter Linkage	Enter Linkage
Call 'SUB' using OVALL1	Entry 'SUB' using OVALL2
Enter COBOL	Enter COBOL

DOS/360 QTAM 2260 LINE ADDRESSING FEATURE

If the Line Addressing Feature of the 2260 (remote) is used, it is necessary that the first byte is the user generated output message in a STX (Start of Text) character. The user must place this STX byte in the Message.

The second byte must be the line addressing byte.

Without the line addressing, the first byte of the user generated message should be a text byte.

The I/O module for a normal write to the 2260 inserts the needed STX byte; for a write to the 2260 with line addressing the needed STX byte is not inserted.

DOS/360 UTILITIES AND JOB CONTROL-END OF CARD INPUT

One feature of the DOS utility programs is frequently overlooked. The utility programs recognize /* as the end of a card input file only if the rest of the card is blank. This ability is essential to build job input streams on tape or disk. Placement of comments (positions 4-80) in all /* statements as standard practice would permit loading of an input stream to tape or disk without modification of the /* statements. The following cards could be used to build a SYSIN tape:

```
// JOB BUILD SYSIN ON TAPE
// ASSGN SYS004,x'00C'
// ASSGN SYS005,x'180'
// UPSI 1010
// EXEC CDTP
// UCT TC,FF,A=(80,80),B=(80,80),OR
// END
// JOB JOB1
     Remainder of JOB1
/* SOME PUNCH
/& SOME PUNCH
// JOB JOB2
     Remainder of JOB2
/* SOME PUNCH
/& SOME PUNCH
/*
/&
```

The /* without any punch in positions 3-80 signals the end of card data for the utility program. The /& performs the usual end-of-job functions.

DOS/360 COBOL ISFMS DATA ON 2321, INDICES ON 2311 CLARIFICATION

An article on the above subject was published in Newsletter 68-02, page 10. Because of a misinterpretation, a "correction" to the article was published in Newsletter 68-03, page 9. The following is a clarification supplied by the author.

In issue 68-03, page 9, the paragraph that reads:

"2. In Exhibit 2, A and B should be defined explicitly as datanames. Additionaly, an A is X'C1', not X'01' and B is X'C2', not 'X02'."

should be deleted.

In the original article in Installation Newsletter 68-02, Exhibit 2, page 12, the non-numeric literal 'A' located in the COBOL sentences,

MOVE 'A' to Byte (22).
MOVE 'A' to Byte (201).

is a X'01' or a binary 0000 0001.

The character 'A' that was printed is the character the 2203 printer with a 52 character set print bar interpreted from the punch card code of a 12, 9, 1 punch.

The non-numeric literal 'B' located in the COBOL sentence,

MOVE 'B' to Byte (66).

is a X'02' or a binary 0000 0010.

The character 'B' that was printed is the character the 2203 printer with a 52 character set print bar interpreted from the punch card code of a 12, 9, 2 punch.

BOS/360 RPG CONSOLE TYPEWRITER ROUTINE

The following is a field contribution which has not been submitted to any formal IBM test. Potential users should first evaluate its usefulness in their own environment.

The BOS-RPG Specifications Manual indicates no support for the Console Typewriter.

The program (Exhibit 1) was written in RPG under BOS and it works.

BOS/360 RPG GETDAT FROM COMMUNICATIONS REGION ROUTINE

The following is a field contribution which has not been submitted to any formal IBM test. Potential users should first evaluate its usefulness in their own environment.

Using BOS-RPG, a user wanted to retrieve the date from the Communications Region rather than have to define a card file just to read a date card.

An Assembler routine was written, assembled (Exhibit 2), and cataloged in the relocatable library under the name GETDAT. It is exited to once at Calculation time as shown on the RPG specification sheet (Exhibit 1).

The routine was assembled under BOS/360 Assembler Version 1, Modification 10.

BOS/360 RPG CONSOLE TYPEWRITER ROUTINE

IBM IBM System/360 Assembler Coding Form

PROGRAM: Sample Program to Show BOS-RPG Support of the Console Typewriter
PROGRAMMER: DATE: PUNCHING INSTRUCTIONS: GRAPHIC: PUNCH: PAGE OF: CARD ELECTRO NUMBER:

STATEMENT

Name	Operation	Operand	Comments	Identification-Sequence
	H016 016			
	FIN	IPE F 80 80	READ01 SYSIPT	
	FTYPE	Ø F 80 80	PRINTERSYSLOG	
	FPRINT	Ø F 80 80	ØF PRINTERSYSLST	
	IIN	AA 01		
	I		1 80 DATA	
	ØTYPE	D 1 01		
	Ø		DATA 80	
	ØPRINT	D 1 01		
	Ø		DATA 80	
/*				
THIS CARD SHOULD BE PRINTED ON THE PRINTER AND TYPED ON THE CONSOLE TYPEWRITER				
/*				

Exhibit 1

BOS/360 RPG GETDAT FROM COMMUNICATIONS REGION ROUTINE

IBM

INTERNATIONAL BUSINESS MACHINES CORPORATION

REPORT PROGRAM GENERATOR CALCULATION SPECIFICATIONS

IBM System/360

Date _____

Program _____

Programmer _____

Punching Instruction: Graphic | Punch

Page

Program Identification

Line	Form Type	Control Level (LO-LR)	Indicators: And / Not	Factor 1	Operation	Factor 2	Result Field	Field Length	Decimal Positions	Half Adjust (H)	Resulting Indicators: Plus / Minus / Zero or Blank; Compare: High 1>2 / Low 1<2 / Equal 1=2	Comments
01	C	*	CALCS. REQD TO GET	DATE	FROM COMMUNICATION			N		REGION	
02	C	*									
03	C	*									
04	C		N25	MOVE	' / / '	DATE	8				
05	C		N25	EXIT	GETDAT						
06	C		N25	SETON						25	
07	C			RLABL		DATE					
08	C	*									
09	C	*									
10	C	*	ANY AVAILABLE INDICATOR		CAN BE USED	INSTEAD			OF	25	
11	C	*									
12	C	*									
13	C										
14	C										
15	C										
	C										
	C										
	C										
	C										
	C										

Exhibit 1

Issue No. 68-05

BOS/360 RPG GETDAT FROM COMMUNICATIONS REGION ROUTINE

```
LOCATN  OBJECT CODE    ADDR1 ADDR2 STMNT   SOURCE STATEMENT   360B AS 309  V1MA

                                   0001 GETDAT   AFILE LIBRARY,RETAIN
                                   0002 GETDAT   START
000000 05F0                        0003 BALR     BALR  15,0
000002                             0004          USING *,15
000002 9012 F02E       00030       0005          STM   1,2,D
                                   0006          COMRG
                                   0007** CHANGE LEVEL 10
000006 4810 0014       00014       0008*         LH    1,20
00000A 5820 F026       00028       0009          L     2,ADATE
00000E D201 2000 1000  00000 00000 0010          MVC   0(2,2),0(1)
000014 D201 2003 1002  00003 00002 0011          MVC   3(2,2),2(1)
00001A D201 2006 1004  00006 00004 0012          MVC   6(2,2),4(1)
000020 9812 F02E       00030       0013          LM    1,2,D
000024 07FE                        0014          BR    14
                                   0015          EXTRN DATE
000028 00000000                    0016 ADATE    DC    A(DATE)
000030 0000000000000000            0017 D        DC    D'0'
000000                             0018          END   BALR
```

Exhibit 2

BPS/360 CATALOGING USER PROGRAMS ON SYSTEMS TAPE

The following is a field contribution which has not been submitted to any formal IBM test. Potential users should first evaluate its usefulness in their own environment.

BPS/360 users may want to take advantage of the stacked job capabilities of an operating system and eliminate the loading of programs via cards. A solution is to catalog problem programs on the BPS systems tape. This offers two advantages over storing programs on separate loadable tapes. First of all, fewer program tapes need be kept in the library. Second, problem programs to be executed and source programs to be compiled can be intermixed without having to change program tapes.

To catalog the user's object programs on the BPS sytems tape:

1. Insert the following cards in the reader:

```
// JOB SYSCMA
// DATE
// ASSGN SYSRDR
// ASSGN SYSIPT     only device assign-
// ASSGN SYSOOO     ments not currently
// ASSGN SYSOO1     in effect are to be
// EXEC             made.
// CATAL
 PHASE NAME, A X'aaaa'   NAME is 1-6
 *                       character name
 * USER                  of program to
 *    PROBLEM            be cataloged;
 *          PROGRAM(s)   'aaaa' is hex-
 *                       adecimal start-
 ENTRY                   ing location of
// END                    program.
```

Any number of programs can be cataloged at one time by placing them (with their associated CATAL, PHASE and ENTRY cards) between the EXEC and END cards, but they must be in alphabetical order by program name.

2. Mount the BPS systems tape on a drive other than that used by SYS000 and SYS001.

3. Ready the card reader and perform an IPL from the drive with the systems tape.

4. At end-of-job, SYS000 contains the expanded systems tape.

To execute programs from this tape:

1. Insert the job control cards for each program in the card reader and ready it. Job name is the same as that used to identify the program on the phase card.

2. Mount the program tape on a drive and perform an IPL from it.

3. Job-to job transition is automatic and continuous. To obtain a halt between jobs, place a // PAUSE card before the next JOB card. This places the system in the WAIT state, enabling the operator to mount new tapes, change paper, etc. To call in the next program, press "55" on the 1052 and the interrupt key on the console.

S/360-20 PURGING INVALIDLY PUNCHED CARDS

The following is a field contribution which has not been submitted to any formal IBM test. Users should first evaluate its usefulness in their own environment.

Exhibit 1 represents the flow chart and machine language coding to produce a one-card program for the S/360 Model 20 which will purge any file of cards containing invalid punches. It requires an MFCM, and will stacker select into pocket 3 all cards which contain any invalid punches. It is extremely helpful with files containing "under-punching" such as those from a Unit-Record account.

1130 CSP-TYPE ROUTINES FOR THE 1403, 2501, AND 1442-5

The overlapped I/O routines of the Commercial Subroutine Package are being used by many 1130 installations. Conversion to the 1403, 2501 and/or 1442-5 may present problems in this area since CSP routines for these devices are not available.

Exhibit 1 shows three subroutines (P1403, R2501 and P1442) which may be used in place of PRINT, READ and PUNCH, respectively. Basically, the routines are slightly modified versions of the CSP routines.

P1403/S1403 will operate in the same way as PRINT/SKIP, except, of course, that the 1403 will be used instead of the 1132. To avoid changing source programs, the user may wish to call the subroutine PRINT/SKIP; in this case, substitute PRINT for P1403 and SKIP for S1403 everywhere these names appear in the Assembler listing. Carriage control, channel skip, and error codes will remain the same as they were for the 1132.

R2501 will operate the same as READ. As mentioned above, the name of the routine may be changed from R2501 to READ-everywhere R2501 appears, change it to READ.

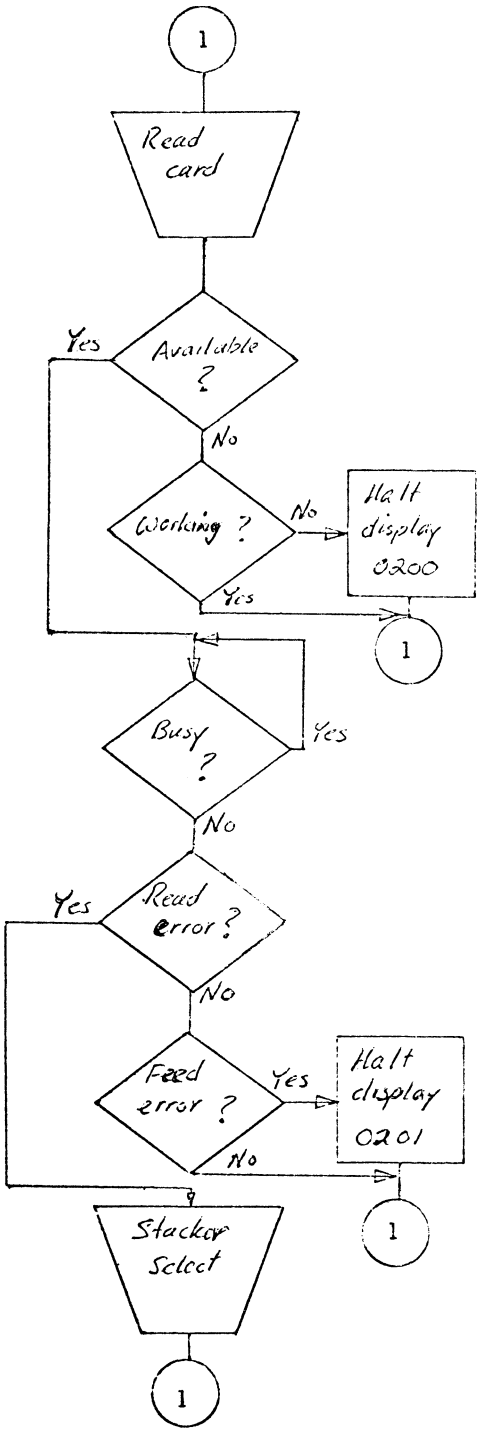
P1442 will punch on the 1442-5. The comments made above also apply to this routine.

If these routines are to be used with Version 1 of the Monitor, several Version 2 Subroutine Library routines must be loaded onto the Version 1 disk:

P1403/S1403	uses	ZIPCO, EBPT3, PRNT3
R2501	uses	READ1
P1442	uses	PNCH1

All use the ARGS subroutine of CSP (whose entry points are ARGS, RPACK and SWING).

S/360-20 PURGING INVALIDLY PUNCHED CARDS




```
D0  22  02  00  00  50
47  80  00  B2
47  40  00  9C
99  00  02  00
47  F0  00  9C
9A  20  00  B2
9A  21  00  C2
9A  25  00  CA
47  F0  00  9C
9B  20  00  03
47  F0  00  9C
99  00  02  01
47  F0  00  9C
```

Exhibit 1

1130 CSP-TYPE ROUTINES FOR THE 1403, 2501, AND 1442-5

```
        ENT     P1442
JCARD   DC      *-*
AREA    BSS     81
FLAG    DC      *-*
P1442   DC      *-*
        STX   1 SAVE1&1
        LDX   I1 P1442
        LIBF    ARGS
        DC      JCARD
        DC      JLAS2
        DC      AREA
        DC      80
        LD      AREA
        STO     CNT2
        LD    1 3
        STO     ERR+1
        SRA     16
        STO     FLAG
        LIBF    SWING
        DC      JCARD
        DC      JLAS2
        LIBF    SPEED
        DC      /0011
JLAS2   DC      *-*
        DC      AREA&1
CNT2    DC      *-*
        LIBF    PNCH1
        DC      /2000
        DC      AREA
        DC      ERROR
        LIBF    SWING
        DC      JCARD
        DC      JLAS2
TEST    LIBF    PNCH1
        DC      /0000
        MDX     TEST
        MDX   1 4
        STX   1 DONE+1
SAVE1   LDX   L1 *-*
DONE    BSC   L  *-*
ERROR   DC      *-*
ERR     STO   L  *-*
        MDX   L  FLAG,1
        BSC   I  ERROR
        END
```

```
        ENT     P1403
        ENT     S1403
ONE     DC      1
SPACE   DC      /2000
JCARD   DC      *-*
JLAST   DC      *-*
AREA    BSS     61
P1403   DC      *-*
        STX   1 SAVE1&1
        LDX   I1 P1403
        LIBF    ARGS
        DC      JCARD
        DC      JLAST
        DC      AREA
        DC      120
        LD      AREA
        A       ONE
        SRA     1
        STO     AREA
        SLA     1
        STO     CNT
        LD    1 3
        STO     ERR&1
        LIBF    RPACK
        DC      JCARD
        DC      JLAST
        DC      AREA&1
        LIBF    ZIPCO
        DC      /0000
        DC      AREA+1
        DC      AREA+1
CNT     DC      *-*
        CALL    EBPT3
TEST    LIBF    PRNT3
        DC      /0000
        MDX     TEST
        LIBF    PRNT3
WRITE   DC      /2000
        DC      AREA
        DC      ERROR
        LD      SPACE
        STO     WRITE
        MDX   1 4
        STX   1 DONE1&1
SAVE1   LDX   L1 *-*
DONE1   BSC   L  *-*
ERROR   DC      *-*
ERR     STO   L  *-*
        SRA     16
        BSC   I  ERROR
S1403   DC      *-*
        LD    I  S1403
        STO     ARG&1
ARG     LD    L  *-*
        BSC   L  NOSUP,-Z
        LD      NOSPC
        STO     WRITE
        MDX     DONE
NOSUP   STO     CNTRL
        LIBF    PRNT3
CNTRL   DC      /3000
DONE    MDX   L  S1403,1
        BSC   I  S1403
NOSPC   DC      /2010
        END
```

```
        ENT     R2501
JCARD   DC      *-*
AREA    BSS     81
FLAG    DC      *-*
R2501   DC      *-*
        STX   1 SAVE1&1
        LDX   I1 R2501
        LIBF    ARGS
        DC      JCARD
        DC      JLAS1
        DC      AREA
        DC      80
        LD      AREA
        STO     CNT1
        LD    1 3
        STO     ERR&1
        SRA     16
        STO     FLAG
        MDX   1 4
        STX   1 DONE&1
        LD      ONE
        LDX   L1 80
MO      STO   L1 AREA
        MDX   1 -1
        MDX     MO
        LIBF    READ1
        DC      /1000
        DC      AREA
        DC      ERROR
CONVT   LIBF    SPEED
        DC      /0010
        DC      AREA&1
JLAS1   DC      *-*
CNT1    DC      *-*
        LD      FLAG
        BSC   L  FINAL,&-
        SRA     16
        STO     FLAG
        MDX     CONVT
FINAL   LIBF    SWING
        DC      JCARD
        DC      JLAS1
TEST    LIBF    READ1
        DC      /0000
        MDX     TEST
SAVE1   LDX   L1 *-*
DONE    BSC   L  *-*
ERROR   DC      *-*
ERR     STO   L  *-*
        MDX   L  FLAG,1
        BSC   I  ERROR
ONE     DC      1
        END
```


Exhibit 1

NEW TYPE III PROGRAMS *

The following are the abstracts of programs which have been recently made available from the Type III library.

The program, along with its complete abstract, will be incorporated in subsequent issues of the Catalog of Programs.

Programs may be obtained by submitting a properly completed "General Program Request Card" (Form Number 120-1145-1) to the Program Information Department, 40 Saw Mill River Road, Hawthorne, New York, 10532.

These programs and their related documentation are distributed by IBM in the author's original form and have not been subjected to any formal testing.

Any discussion of Type III Programs must emphasize the following points:

1. **Type III programs are not part of the IBM product line.** This is a Type III program.
2. **Type III programs have not been subjected to any formal product test.**
3. **Recipients of Type III programs are expected to make the final evaluation as to the usefulness of the programs in their own environment.**
4. **There is no committed maintenance for Type III programs. However, any changes the author chooses to make will be announced in subsequent issues of the Catalog of Programs.**

* **NOTE: THE CUSTOMER MUST BE INFORMED THAT THE ABOVE APPLIES TO ANY OF THE FOLLOWING TYPE III ABSTRACTS FURNISHED TO HIM.**

OSDEBE - S/360 OS CARD/PRINTER/TAPE MULTIPLE UTILITY PROGRAM. This System/360 program is an adaptation of the original BPS DEBE utilities program to operate under the Operating System. It requires an online card reader/punch and a SYSOUT writer. This program performs all of the functions provided by the BPS DEBE without the necessity of OS Data Definition (DD) statements for the devices utilized. Only one DD card for the SYSOUT writer is required. Source program written in Assembler.

Ordering Procedure: Order File Number 360D-00.1.015.

The program material can be obtained in card form or on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

The requester must indicate whether a 9 track or 7 track DTR is required. If not specified, a 9 track DTR will be provided.

No tape submittal is required. The DTR will be provided by the Library.

GENERALIZED OS/360 ASSEMBLER LANGUAGE INPUT/OUTPUT SUBPROGRAM FOR USE WITH ASSEMBLER LANGUAGE AND/OR PL/I PROGRAMS. An OS/360 Assembler Language Input/Output Sub-program capable of performing many commonly required I/O operations on sequential, Index-sequential and partitioned data sets. The I/O package supports user modules written in Assembler Language, PL/I or a Bi-Lingual environment where both BAL and PL/I user modules are present. Any Number of data sets can be managed in a single execution. User programs invoke the I/O Subsystem by means of call statements and parameter lists. I/O Base Modules are combined with user programs through Linkage Editor processing. Advantages and features of the I/O package are summarized as follows:
1. Simplifies and standardizes I/O programming for the application programmer.
2. Provides a common interface between all user programs and the user data base.
3. Provides error editing and error reporting facilities. 4. Provides an ISAM Journal and unprocess capability. 5. Performs utility functions such as record count accumulation and display, DCB Format Dumps, DSCB Format Dumps, CVT Format Dumps.

Ordering Procedure: Order File Number 360D-04.3.003.

The program material can be obtained on one 9 track or 7 track (Data Conversion feature requied) Distribution Tape Reel (DTR).

The requester must indicate whether a 9 track or 7 track DTR is required. The DTR will be provided by the Library.

S/360 HIGH-WATER-MARK. This is a program designed to determine the largest size of a problem program running under the Operating System with or without Option 2. It becomes part of the control program nucleus in the form of a Type I SVC. Its function is to intercept each GETMAIN request, calculate the size of the partition, problem program, and largest free area; and when the problem program is at its largest, store the information in the nucleus. A service routine, such as the accounting routine, must be invoked in order to print the information. It will be approximately seven hundred bytes in size.

Ordering Procedure: Order File Number 360D-04.4.006.

The program material can be obtained in card form or on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

The requester must indicate whether a 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

The requester must indicate whether a 9 track or 7 track DTR is required. If not specified, a 9 track DTR will be provided.

No tape submittal is required. The DTR will be provided by the Library.

S/360 BASIC AUTODIN ACCESS METHOD. This access method provides the AUTODIN user with macros and associated supervisor functions necessary to interface his message control and processing programs with the AUTODIN System. The program is used with DOS/360 and is written in BAL. A simulator module is provided to allow user testing prior to installation of the 2701 AUTODIN Adapter hardware.

Ordering Procedure: Order File Number 360D-06.3.003.

The program material can be obtained on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

The requester must indicate whether a 9 track or 7 track DTR is required. If not specified, a 9 track DTR will be provided.

No tape submittal is required. The DTR will be provided by the Library.

BASIC AUTOMATIC NUMBERING TESTING AND MAINTENANCE (BANTAM). The BANTAM Update Program is a utility written in Assembler language to operate under DOS/TOS Supervisors. The utility will maintain source libraries on disk (DOS only) or tape with facilities to:

1) add, delete, number and name members.
2) add, delete, and change source statements.
3) create input to compilers.
4) generate job control statements to create input files for a catalog.

Use of BANTAM options in a testing environment enables the user to establish an effective testing procedure, eliminating inefficient source and object deck handling. BANTAM interfaces with DOS/TOS library and compiler facilities to give the user maximum system utilization.

Use of this program in a testing environment will offer the user:

1) an ordered file of his "in-test" and production programs.
2) the elimination of unnecessary source and object deck manipulation.
3) documentation of all changes made to his programs.
4) the ability to keep job control statements for compilation, linkage ed-

iting and execution of a program in a private library with the source statements.

5) The facility to compile, linkage edit and execute or to compile and catalog without card handling.

SYSTEM CONFIGURATION

The BANTAM Update Program requires the basic machine requirements outlined in the System Control and System Service Programs manual for DOS and TOS respectively.

In addition, to operate BANTAM in a TOS environment, the following requirements must be met:

32K bytes of main storage
three tapes for BANTAM Update Phase
five tapes for operating in the BANTAM Test Environment.

To operate BANTAM in a DOS environment, the following basic requirement must be met:
32K bytes of main storage

Ordering Procedure: Order File Number 360D-06.7.008.

To obtain program material, submit one 2400 foot reel of magnetic tape. Specify whether 7 or 9 track recording is required. If not specified, 9 track recording will be used.

The required tape may be ordered from IBM or supplied with your request for the program.

MANUFACTURING RECORDS PROCESSOR FOR IBM OPERATING SYSTEM/360. Manufacturing Records Processor can speed up and help optimize Production and Inventory Planning in many manufacturing and service organizations. It does this by creating and reading up to four different files on DASD. These files are Parts Master, Product Structure, Operations Routing, and Workcenter Master; they are arranged so that any change need be made to only one file, thus obviating the possibility of unwanted different change levels existing at one time. Full flexibility is available by allowing for code, easily-written by the user, to be incorporated. This could permit several engineering change levels or alternate production facilities to exist at one time for any part or assembly. Manufacturing Records Processor is a set of generalized programs and macros which enables the user to design, build, and execute his own programs under IBM/360 OS.

The Manufacturing Records Processor is an adaptation of the Bill of Material Processor Version II operating under BOS/DOS (360A-ME-06X).

M/C Configuration

System/360 Model 30 and up with Decimal Arithmetic.

65K or 128K core storage depending on features used.

Minimum one disk drive in addition to one or two required for resident OS: Printer, Card Read/Punch.

Operating System Requirements

The Manufacturing Records Processor Program is designed to operate under any version of Operating System/360 with some special requirements which are discussed in the following sections.

Required OS Features

The OS configuration required for use of the MRP package is any OS system with the additional requirements: 1) the index sequential access method must be included at SYSGEN time because of the method of organization used for the master files, and 2) COBOL must be available if the user wishes to use it for his retrieval programs or if he runs the entire sample problem.

Ordering Procedure: Order File Number 360D-23.3.005.

The program material can be obtained on one 9 track magnetic tape. Submit one reel of magnetic tape. The required tape may be ordered from IBM or supplied with your request for the program.

1130 PRINTER CONTROL PROGRAM FOR THE 1132 FOR OPERATION UNDER THE 1130 DISK MONITOR. The purpose of the Printer Control Package (PCP) is more efficient usage of the 1132 Printer. It is a two part package of assembler written programs and can provide a maximum increase of 67 percent in print speed.

The first part of the package is a modification of the IBM Type I PRNT1 Printer Control Program. The modifications include a scan routine which ends the print operation after the last character is printed. An alpha only list will aproach 110 lines per minute.

The second part of the package represents a complete set of inter-related control programs. It offers the improvements of the modified PRNT1 in addition to a system of buffer and queue management. The routines are FORTRAN callable and are used in place of PRINT, SKIP, WRTS, WRTD and SKSP.

Ordering Procedure: Order File Number 1130-03.4.005.

Distribution will be in card form only.

1130 PACIN-PROGRAM FOR THE ANALYSIS OF CAPITAL INVESTMENT. This program is an adaptation of "Capital Investment Program #1620-CS-01X" to the IBM 1130 computer. It is designed to accept various input parameters and data, and from this compute tables of depreciation and salvage values, earnings and interest payments. From these tables a stream of cash flow is generated and this in turn is used to compute a table of rates of return. It is written for an 8K 1130 system with an 1132 printer. The language is FORTRAN.

Ordering Procedure: Order File Number 1130-19.0.004.

Distribution will be in card form only.

1800 STORAGE CRT DISPLAY SYSTEM. The 1800 Storage CRT Display System consists of a set of subroutines to be used for generation of displays on storage oscilloscopes interfaced to an 1800 System through the digital-to-analog converters and electronic contact operate features. These subroutines enable generation of characters for text display, scaling and plotting of vectors for waveform or graph display, conversion and display of floating point numbers, and plotting of axises for graphs.

The user must assemble equate cards punched with values to specify his hardware configuration, common areas, masking requirements, display area size definition, and character and line size definition.

The routines are written to support a variable number of displays up to sixteen, but can be easily modified to expand this maximum value. The ECO points for each display must all be in the same ECO group. Only two DAC's with the buffer register feature are required.

The subroutines are written in Assembler Language and are provided in source form only.

Prdering Procedure: Order File Number 1800-03.4.006.

Distribution will be in card form only.

1401 MATCH AND MODIFY FILES PROGRAM. The Match and Modify Files Program is a self-modifying program which will process any tape file against any other tape file (subject to size limitations) comparing records from one file to records on the other, including the ability to table-search one of the input files. Based on the comparisons and/or table search the program can generate two output files in the same format as the input or reformatted as the user desires. The program requires a 16K 1401 or 1460 with four 729 tape drives, 1402 card reader, a 1403 printer with 132 print positions, Advanced Programming and High-Low-Equal Compare. It is written in 1401/1460 Autocoder language and has been assembled on the 1460 Disk Autocoder Processor.

Ordering Procedure: Order File Number 1401-01.4.231.

The program material can be obtained in card form or on one 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

No tape submittal is required. The DTR will be provided by the Library.

1401 GENERAL PROCESSING UTILITY PROGRAM. The GENERAL PROCESSING UTILITY PROGRAM performs many simple utility jobs often called for in a computer installation and is intended to reduce the number of individual utility programs that have to be maintained each of which performs only one or a few of these jobs. It does card to card, card to printer, card to tape, tape to tape, tape to printer, and tape to card operations in a total of twenty-six options each of which is called for by a combination of sense switch settings. The program is card or tape loadable and is self-initializing between jobs without re-loading. The program requires an 8K, 12K or 16K 1401 or 1460 with at least one, and up to five, tape drives, a 1402 card reader and punch, a 1403 printer with 132 print positions, and High-Equal-Low compare. It is written in 1401/1460 Autocoder Language and has been assembled on the 1460 Disk Autocoder Processor.

The program material can be obtained in card form or on one 7 track Distribution Tape Reel (DTR).

No tape submittal is required. The DTR will be provided by the Library.

7094 CHAIN OPTIMIZATION FROM CHARACTER USAGE STATISTICS. This program provides a general method of determing the printing speeds of chain (or train) printers for any chain-character configuration from character usage statistics. An optimization and chain layout feature may be used to determine an optimum chain configuration for given types of printing jobs, often resulting in significant increases in printing speeds. In order to be useful the printer must have a capability similar to that provided by the Universal Character Set Feature. Input to the program consists of character usage statistics (from any number of different applications) printer hardware information (number of characters on chain number of characters per slug, speed limitations) and, if desired, slug defintions and/or chain layout.

The output generally consists of the optimum arrangement (including a layout schematic) and the corresponding printing speed.

The program was primarily written in FORTRAN IV and operates under Version 13 of IBSYS, Running time on an IBM 7094 computer is usually 2 or 3 minutes. A standard 7094 system configuration is required.

Ordering Procedure: Order File Number 7094-SDA-3542.

Submit one 2400 foot reel of magnetic tape. The required tape may be ordered from IBM or supplied with your request for the program.

IBM Installation Newsletter

February 23, 1968 Issue No. 68-03

CONTENTS

Distribution: Branch Offices - DP Management, Salesmen, Systems Engineers, FE Managers, Regions, Districts, Education Centers, Field Systems Centers, Federal Systems Centers, FE Area Offices, DPD HQ, FED HQ, WTC.

*Required Immediate Attention

For IBM Internal Use Only

DPD HQ, White Plains, N. Y.

SELECTED SHORT SUBJECTS

The purpose of Selected Short Subjects is to bring together and highlight concise, factual and timely information which will indicate that action is to be taken by the IBM representative whose accounts are affected.

1. S/360-20 MFCM Effect of Improperly Scored Cards

The following article is intended to improve the customer's satisfaction with the 2560 MFCM through the use of acceptable card scores.

Field Engineering advises that they are experiencing considerable difficulty in maintaining installed 2560's (MFCM) on the Model 20 due to the attempted processing of improperly scored cards.

The only card scores acceptable for processing on the 2560 are:

1. Internal Scores (Before Separation)

M	-	4
OM	-	3
S	-	1
M	-	5 (only 1000 cards permitted in stacker)

2. External Scores (After Separation) for Reading and Punching

Column 80 End

M	-	3
M	-	5
M	-	7
CF	-	4

3. Column 1 End

M	-	7

4. For Reading Only Column 80 End

M	-	4
M	-	6
OM	-	2

2. S/360-20 EC Affecting User Written Console Routines

The following information has been previously released in RETAIN message 2030/031; also, 2030 ECA #323 describes EC 128064. It is published here as a reminder that the change is being made.

The System/360 architecture specifications state that the Incorrect Length Bit will be set in the channel status byte for any write operation to a device. If this status is not desired, the Suppress Incorrect Length Indication bit must be set in the CCW to mask it off.

Until recently, the S/360 Model 30 has not complied with these rules if the console typewriter was the device involved. The 2030 did not present an ILI on write operation to the 1050 console. Some customers may have written their own console routines. Many of these routines do not use the SILI Bit, and due to 2030 operation, there were no negative effects.

However, EC 128064 with concurrent EC 126753 is now being retrofitted to all 2030's to bring this machine into line and the 1050 console will then present an ILI on all write operations.

Those customers who have written their own console physical IOCS routines should set the SILI on in all output CMDS. This will prevent the output chain from being broken by the ILI condition, causing part of the output message to be lost.

OS/360 RELEASE 15 PLANNING INFORMATION - GENERAL

The following information is provided to you for planning purposes only. The actual availability of all items will be announced in Program Announcement Letters, the official IBM announcement media.

The announced schedule for OS/360 Release 15 items is July 31, 1968. An assessment of the present status of the Release 15 integration and testing indicates a possibility that the release may be available from PID as early as April, 1968. Release 15 is now projected

to include the items listed below in addition to maintenance. The items marked with asterisks are subject to last minute withdrawal if unforeseen problems are encountered in final testing.

1. MFT (Version II) as described in the planning SRL Planning for Multiprogramming with a Fixed Number of Tasks-Version II (C27-6939-0).

*2. Data Set Integrity Features with MFT-II (Shared or Exclusive use parameters in DD statements) as now available under the MVT option.

3. Rollout/Rollin support under MVT as described in the SRL Planning for Rollout/Rollin (C27-6935-0).

* 4. Some Additional Improvements to MVT which are considered as incremental improvements. These include:
   a. An improvement to the PCI fetch which will permit faster program loading when programs reside on the 2301 drum.
   b. The job class capability (similar to that described for MFT-II).
   c. Support for the 2250 Display Unit as a console device, in conjunction with the present 1052 console.
   d. User exit at step initiation, similar to that of MFT and PCP.
   e. Use of cataloged procedures for initiators provides more flexibility and reduces the possibility of error.

* 5. COBOL F support of the Asynchronous Processing feature, including the following:
   a. APPLY RECORD PROTECTION ON for files whose organization is direct or random and whose access is random.
   b. APPLY section-name TO saved-area-name file-name FOR integer CYCLES.
   c. The USE FOR RANDOM PROCESSING declarative.
   d. The HOLD statement.
   e. The PROCESS statement.

6. Linkage Editor F (Version II) as announced in Letter P68-11.

7. Recovery Management support for the Model 65 as announced in Letter P68-2 (B).

*8. PL/I F (Version IV). The key features included are: Locate mode record I/O with the ability to read self-defining records and overlay defining; a major subset of the list processing facilities; a major subset of the tasking facilities (the user should carefully analyze the supporting documentation when available); an option to read and write some COBOL data sets; PUT-DATA without a data list. A number of other operational, functional and performance improvements are also provided.

*9. A Checkpoint/Restart facility improvement which permits the checkfile or SYS1.SYSJOBQE to be on the 2314 and permits the use of dual density tape drives for both program use and for the checkfile. In addition, input/output operations will be automatically quiesced at checkpoint time for the task requesting a checkpoint (formerly it was the user's responsibility to assure that all I/O operations had been stopped).

*10. Support for the 2844 control unit on a single CPU which permits two access paths to each of the 2314's disk packs.

*11. IOS priority queueing facility. With this feature, designated I/O

devices will be given the priority of the task requesting the I/O operation. This is particularly designed to permit jobs such as remote inquiry to be given faster access I/O devices to improve response time.

12. Improvements in teleprocessing support including the following:
    a. BSC support under BTAM for the 2780 operating in 6 bit mode.
    *b. Dial support under QTAM is improved to permit two-way communication for terminal initiated calls, eliminate the need for queue priming busy lines and to permit the "converse mode" to be used on any line in a multiple line communication group.

*13. An improvement to the system generation procedure which adds new parameters to the GENERATE macro instruction. These allow the user to control the disposition of utility data sets used during system generation and permit more flexibility in naming and space allocation for the work area required by the IEHMOVE program in Stage II.

*14. An improvement in the initial program load procedure (NIP) which reduces the number of key strokes necessary by the operator. Permits the use of lower case characters and allows the use of commas as parameter delimiters.

*15. The Error Recovery transient area is expanded from 400 bytes to 1024 bytes to permit both functional and performance improvemnts. The tape error recovery procedure routines are improved and support is provided for the Channel Check Handler (CCH) portion of Model 65 Recovery Management support.

OS/360 - RELEASE 14 WITH THE 2870

OS/360 Release 14 contains a change to IOS to eliminate the "Partitions Monopoly" condition. This change may cause a slight reduction in the amount of overlap between selector subchannels and the multiplexor subchannels of the 2870.

The IOS changes are included in the Test Channel Modules and consist of the issuance of a TCH command if the unit address to be started is Hex 80 or greater, i.e. ØCØ-OFF.

This will result in any pending interrupts on channel zero being serviced by IOS before the new request is started.

Reduction in overlap occurs only when the channel zero pending interrupt belongs to a selector sub-channel not equal to the new request. Example, Pending ØD1, but new request ØC5.

No data is available as to the reduction incurred. It is entirely dependent upon the number of devices attached to the 2870 and the degree of multiple buffering used.

DOS/360 BASIC PL/I-WAIT STATE WITHOUT DIAGNOSTIC OUTPUT

A PL/I program which exceeds core capacity by somewhat less than the size of the last CSECT processed by LNKEDT may fail during execution when the last CSECT is executed. No diagnostics are printed at LNKEDT or during execution and the system goes into an unrecoverable Wait/System State apparently trying to access core beyond capacity. Recovery can be effected by reducing the PL/I program by about 200 Bytes, dependent on the size of the last CSECT.

S/360-20 DISK RPG MULTIPLE CHAINING TO SAME FILE

The following is a field contribution which has not been submitted to any formal IBM test. Potential users should first evaluate its usefulness in their own environment.

1. The S/360-20 SRL's indicate that you cannot chain to a file more than once from the same record, or

from a file to itself, in S/360-20 Disk RPG.

2. The chaining described above can be accomplished by defining the chained file more than once in the RPG program, each time with a different file name.

3. Additional VOL, BLAB, and X-TENT control cards are required for successive definitions of the file. They are identical to those normally used for the file, with the exception of the file name in the VOL card.

4. Updating a multiply-chained file as a result of each chaining can cause errors and should be avoided.

SYSTEM/360 ISOLATION FEATURE

Product Announcement Letter 267-41 details this no charge feature whose purpose is to eliminate "spurious noise in the system".

The isolation feature is now available for field or plant installation. Those accounts using direct control, channel to channel adapter, control units equipped with two channel switch features, or inter-connected CPU's through channel switches should ensure that they have this no charge feature.

The isolation feature is a hardware modification which does not require any programming changes.

For further information, see Product Announcement Letter 267-41 and the Sales Manual, Machines section, under the processing unit involved.

TELEPROCESSING DESIGN

The IBM Distribution Center has available two manuals that will help in designing and implementing Teleprocessing systems. These are World Trade publications and are not new, but can be used for guidance in understanding the subjects. They are not available to customers.

1. Z10-003 Handbook for the Design of Small Communications-based Systems, which contains detailed information on: 1050, 2740, 2260 Display Terminals and S. T. R. Devices as terminals to a System/360.

2. Z10-0004 2780 Data Transmission Terminal World Trade Marketing Guide, which contains a detailed description of 2780 operation, as well as a comprehensive description of all standard and special features. This guide also provides graphs for all possible transmit and receive devices at speeds of 1200 to 2400 bits per second.

INFORMATION ABOUT THE NEWSLETTER

EVERY SALESMAN AND SYSTEMS ENGINEER SHOULD RECEIVE ONE COPY OF THE NEWSLETTER. Each FE Branch Manager should receive five copies for distribution to Customer Engineers.

The IBM Installation Newsletter is distributed only to IBM locations and is not available to customers. The Newsletter is separated into two parts. The first part contains information which is intended only for IBM personnel and the pages are labeled for IBM Internal Use Only. THESE PAGES, OR REPRODUCTIONS OF THEM, ARE NOT TO BE GIVEN TO CUSTOMERS. However, using your own good judgement, you may discuss pertinent information from these pages with a customer. The second part contains information which may be REPRODUCED by the Branch Office at their discretion and given to customers. These pages are not labeled with any restrictive classification. Note that when any abstracts for Type III programs are given to the customer, he must be informed that the disclaimer for Type III programs, as contained in the Newsletter, applies. YOUR COOPERATION IN PROPERLY HANDLING THE CONTENT OF THE IBM INSTALLATION NEWSLETTER IS NECESSARY FOR ITS CONTINUED PUBLICATION.

Changes to the quantity of the Newsletter shipped are to be made by the Administration Manager. (NOTE THAT EXTRA COPIES OF THE NEWSLETTER ARE NOT TO BE ORDERED FOR DISTRIBUTION OF INFORMATION TO CUSTOMERS.) He should update and send the DP Administration Managers' Communication Control card (form number N44-780) to DP Communication Control, DPD HQ.

Direct inquiries concerning the contents of this Newsletter to:

W. P. Gaffigan, Editor
IBM Installation Newsletter
Installation Planning, Department #713
DPD HQ
112 East Post Road
White Plains, New York 10601
Telephone 914 WH9-1900 Extension 4781

NEW PUBLICATIONS ANNOUNCED BY PRL'S

The weekly PRL's (Publications Release Letters) are used to insure that all Salesmen and Systems Engineers are aware of new or revised Marketing Publications. Normally, each issue of the Newsletter will contain information from two PRL's, one following the other. The information will be placed in the Newsletter in its original form with no rearrangement of form numbers or titles. It is not intended to replace existing information and distribution sources. You should be certain that you are aware of these sources.

PRL #4 January 26, 1968

FORM NO.	TITLE	DISPOSITION
320-0904-6	Obsolete Replace List Effective January 12, 1968	Scrap 320-0904-5
C20-1601-7	Catalog of Programs for IBM 1240-1401-1420-1440 and 1460 DPS - December 1967	Scrap C20-1601-6 & N20-0013-12
Z20-0100-18	IBM Publications and Programs Current Price List	Scrap Z20-0100-17
	MARKETING PUBLICATIONS	
520-2036-0	IBM S/360 Model 20 and IMPACT for Wholesale Distributors	NEW
545-0055-1	PBX Specification Sheet	Use 545-0055-0
C24-9010-1	IBM S/360 Model 20 Tape Programming System Performance Estimates	Use C24-9010-0, N33-8518 & N33-9005
C24-5064-1	IBM S/360 TOS Programming Index	Scrap C24-5064-0
C26-3723-0	IBM 1800 Time-Sharing Executive System Subroutine Library	NEW
C26-5929-4	IBM 1130 Subroutine Library	Scrap C26-5929-3
C27-6912-5	IBM S/360 Operating System Graphic Programming Services for IBM 2260 Display Station Local Attachment	Scrap C27-6912-4
C28-2037-1	IBM S/360 Time Sharing System System Messages	Scrap C28-2037-0
C28-6539-5	IBM S/360 Operating System: Job Control Language	Scrap C28-6539-4, N28-2214, N28-2226, N28-2242 & N28-2220
C28-6586-6	IBM S/360 Operating System: Utilities	Scrap C28-6586-5 & N28-2264, N28-2225
C28-6602-2	IBM S/360 Operating System: FORTRAN IV (H) Programmer's Guide	Scrap C28-6602-0, -1, N28-2211, N28-2228 & N28-2247
C28-6603-2	IBM S/360 Operating System: FORTRAN IV (E) Programmer's Guide	Scrap C28-6603-1, N28-0211, N28-0233 & N28-0235
C28-6631-3	IBM S/360 Operating System: Messages and Codes	Scrap C28-6631-1, -2, N28-2235, N28-2250, N28-2258 & N28-2277
C28-6639-1	IBM S/360 Operating System FORTRAN IV (G) Programmer's Guide	Scrap C28-6639-0, N28-2212, N28-0212, N28-0234 & N28-0236
C28-6656-1	IBM S/360 Operating System: Checkpoint/Restart	Scrap C28-6656-0
C28-6670-0	IBM S/360 Operating System: Programmer's Guide to Debugging	NEW
C28-6671-0	IBM S/360 Operating System Model 65: Shared Main Storage Multiprocessing	NEW
C30-2008-0	IBM S/360 OS; Planning for Remote Entry (MFT Version II)	NEW
E20-0188-3	Retail IMPACT - Inventory Management Program and Control Techniques Application Description	Use E20-0188-2
J20-0003-1	QUIKTRAN Statistical Programs	Scrap J20-0003-0
K20-0248-0	System/360 at Rockland State Hospital	NEW
N20-1071	TNL for Optimum Bond Bidding (360A-FI-06X) User's Manual Re: E20-0228-0	NEW
N24-5299	TNL to DOS Operating Guide BTAM 1287 Support Re: C24-5022-3	NEW
N24-5344	TNL to IBM S/360 DOS Supervisor and Input/Output Macros Re: C24-5037-2 & -3 with -2 N24-5306, N24-5285 and N24-5329	NEW
N24-5346	TNL to DOS Performance Estimates Re: C24-5032-3	NEW
N24-5347	TNL to IBM S/360 DOS System Generation and Maintenance Re: C24-5033-3 with N24-5309, N24-5324, N24-5320 & N24-5266	NEW
N24-5351	TNL to DOS System Control and System Service Programs Re: C24-5036-1 & -2 with -1, N24-5218, N24-5279, N24-5289 & N24-5330	NEW
N26-0209	TNL to 1800 Physical Planning Re: A26-5922-4 with N26-0194	NEW
N26-0211	IBM 1971/1980 TNL Re: A26-3699-0	NEW
N26-0567	IBM S/360 Operating System Assembler (F) Programmer's Guide Re: C26-3756-2	NEW
N27-1287	IBM S/360 Operating System Multiprogramming with a Fixed Number of Tasks (MFT) Concepts & Considerations Re: C27-6926-0	NEW
N28-0229	System/360 Operating System COBOL (E) Programmer's Guide Re: C24-5029-2	NEW
N28-0578	IBM S/360 Operating System: FORTRAN IV (E) Programmer's Guide Re: C28-6603-2	NEW
N28-0579	IBM S/360 Operating System: FORTRAN IV G Programmer's Guide Re: C28-6639-1	NEW
N28-2273	IBM S/360 Operating System: Operator's Guide Re: C28-6540-5	NEW
N28-2281	IBM S/360 Operating System: Storage Estimates Re: C28-6551-4 with N28-2265 & N28-2280	NEW
N28-2282	IBM S/360 Operating System: System Generation Re: C28-6554-3 with N28-2269	NEW
N28-2283	TNL IBM S/360 OS, System Programmers Guide Re: C28-6550-3	NEW
N28-2286	IBM S/360 Operating System: System Control Blocks Re: C28-6628-1 with N28-2262	NEW
N28-2294	IBM S/360 Operating System: Job Control Language Charts Re: C28-6632-1	NEW
N28-2296	IBM S/360 Operating System: Supervisor & Data Management Macro-Instructions Re: C28-6647-0, with N28-2217, N28-2223, N28-2248 & N28-2251	NEW
N28-2297	IBM S/360 Operating System: Supervisor & Data Management Services Re: C28-6646-0 with N28-2202, N28-2246, N28-2263 & N28-2279	NEW
N28-2298	IBM S/360 Operating System: Utilities Re: C28-6586-6	NEW
N28-2306	IBM S/360 Operating System: Operator's Guide Re: C28-6540-5	NEW
N28-2307	IBM S/360 Operating System: FORTRAN IV (H) Programmer's Guide Re: C28-6602-2	NEW
N28-3010	IBM S/360 Time Sharing System Terminal User's Guide Re: C28-2017-1	NEW

For IBM Internal Use Only

PRL #4 (Continued)

N28-3011	IBM S/360 Time Sharing System FORTRAN Programmer's Guide Re: C28-2025-0	NEW
N30-2504	IBM S/360 Operating System Telecommunications Access Method Re: C30-2004-1 with N30-5010	NEW
N30-2505	IBM S/360 Operating System Planning for Remote Job Entry (MFT Version II) Re: C30-2008-0	NEW
N30-5019	TNL to DOS QTAM MCP Re: C30-5004-1	NEW
N30-5024	TNL IBM S/360 Disk Operating System Basic Telecommunications Access Method Re: C30-5001-4	NEW
R20-4082-0	IBM Communications Systems Design & Analysis Education Guide	NEW
R20-8005-1	IBM 1620 Monitor I Prog. System Education Guide	Use R20-8005-0
R20-9163-0	IBM Communications Systems Design and Analysis Course Description	NEW
R20-9221-0	IBM Branch Office Library Course Description	NEW
R29-0111-1	IBM S/360 PL/I Coding P.I. Text	Scrap R29-0111-0
R29-0138-1	IBM PCDPP P.I. 82 Sorter Exam	Use R29-0138-0
R29-0210-2	IBM S/360 COBOL - Writ. Prog. in COBOL P.I. Text	Use R29-0210-1
R29-0211-3	IBM S/360 COBOL - Writ, Prog. in COBOL P.I. Reference Handbook	Use R29-0211-2
R29-0232-3	IBM S/360 ALC Standard & Decimal Instructions Text	Use R29-0232-2
R29-0233-2	IBM S/360 ALC P.I. Appendix	Use R29-0233-1
X20-1760-0	Examples of Control Cards for S/360 Reference Card	NEW
Y24-5018-2	IBM S/360 Tape Operating System Logical IOCS	Scrap Y24-5018-1
Y24-5056-0	IBM S/360 Disk & Tape Operating Systems On Line Test Executive Program	NEW
Y24-5071-0	TNL to IBM S/360 DOS Logical IOCS PLM Re: Y24-5020-2	NEW
Y27-7153	TNL Re: Y27-7113-2 IBM S/360 Operating System Graphics Access Method	NEW
Y27-7157	IBM S/360 Operating System Control Program with Y27-7128	NEW
Y28-2278	IBM S/360 Operating System/360: Input/Output Supervisor, PLM Re: Y28-6616-2	NEW
Y28-2289	IBM System/360 Operating System: Utilities, PLM Re: Y28-6614-2	NEW
Y28-3051	IBM S/360 Time Sharing System - System Logic Summary PLM Re: Y28-2009-0	NEW
Y28-3055	IBM S/360 Time Sharing System Access Method PLM Re: Y28-2016-0	NEW
Y28-6613-2	IBM S/360 Operating System: Job Management, PLM	Scrap Y28-6613-1 & Y28-
Y28-6642-1	IBM S/360 Operating System: FORTRAN H Compiler, PLM	Scrap Y28-6642-0 & Y28-223
Y28-6659-1	IBM S/360 Operating System: MVT Supervisor, PLM	Scrap Y28-6659-0 & Y28-225
Y28-6660-1	IBM S/360 Operating System: MVT Job Management PLM	Scrap Y28-6660-0 & Y28-222
Y28-6665-0	IBM S/360 Operating System: Guide to Model 91 Support, PLM	NEW
Y28-6672-0	IBM S/360 Operating System: Checkpoint/Restart, PLM	NEW
Y28-6819-0	IBM S/360 FORTRAN IV (E) Compiler PLM Re: Y28-6601-2 with Y28-6383	NEW
Y28-6820	IBM S/360 Operating System: FORTRAN IV (G) Compiler PLM Re: Y28-6638-0 with Y28-6384, Y28-6386 &Y28-6388	NEW
Y30-2001-0	IBM S/360 Operating System: Basic Telecommunication Access Method, PLM	Scrap Y30-2001-1 & Y30-2004
Y30-5004	IBM S/360 Disk Operating System Basic Telecommunications Access Method Re: Y30-5001-2	NEW
Z20-1869-0	Laboratory Automation Based (LAB) System	NEW
Z28-6635-3	IBM S/360: CLEAR Operating and Control Procedures	Scrap Z28-6635-2
Z77-7291-0	A Consolidated Group Insurance System	NEW

PRL #5 February 2, 1968

FORM NO.	TITLE	DISPOSITION
320-1621-17	KWIC Index to Marketing Publications/January 12, 1968	Scrap 320-1621-16
N20-1001.38	SRL Newsletter Teleprocessing Re: A24-3089-5	Scrap N20-1001.37
N20-1002.21	DPT Newsletter Re: F20-8172-5	Scrap N20-1002.20
Y20-0073-3	FE Systems Sequence Listing Effective January 12, 1968	Scrap Y20-0073-2
Z20-1720.21	Sales & Systems Guide Re: Z20-1708-1	Scrap Z20-1720.20

MARKETING PUBLICATIONS

221-0476-0	IBM 2420 Magnetic Tape Unit, Model 7 Proposal Insert	NEW
221-0477-0	IBM S/360 Model 85 Proposal Insert	NEW
270-0011-4	9949 Illustrated Parts Catalog	Scrap 270-0011-3
270-0066-1	9955 Physical Planning Manual	Scrap 270-0066-0
320-1906-12	In Brief, December 1967	Scrap 320-1906-11
520-2022-0	LeGendre Polyynomial Promotional Brochure	NEW
520-2023-0	Gamma Function Promotional Brochure	NEW
520-2024-0	Fourier Evaluation Promotional Brochure	NEW

For IBM Internal Use Only

	520-2025-0	Polynomial Evaluation Promotional Brochure	NEW
	520-2026-0	Polynomial Roots Promotional Brochure	NEW
	520-2027-0	Polynomial Fitting Promotional Brochure	NEW
	520-2028-0	Simultaneous Equations Promotional Brochure	NEW
	520-2029-0	Eigenvectors and Eigenvalues Promotional Brochure	NEW
	520-2030-0	Matrix Operations Promotional Brochure	NEW
	520-2031-0	Solution of Nonlinear Simultaneous Equations Promotional Brochure	NEW
	520-2032-0	Forward Difference Method Promotional Brochure	NEW
	520-2033-0	Polynomial Calculation Promotional Brochure	NEW
	520-2034-0	Digital Analog Simulation Promotional Brochure	NEW
	520-2035-0	Zero Wear Analysis Promotional Brochure	Abstract only
	520-2036-0	IBM S/360 Model 20 and IMPACT for Wholesale Distributors Promotional Brochure	NEW
	520-2054-0	IBM S/360 Model 85	NEW
	520-2059-0	IBM 2420 Magnetic Tape Unit, Model 7 Promotional Brochure	NEW
	520-2066-0	IBM Sorters (82/83 Series 50 Folder) Promotional Brochure	Scrap 545-0045-2
	545-0045-3	IBM Executary 224 Operating Manual	Use 545-0053-0
	545-0053-1	SRN Specification Sheet	NEW
	549-0056-0	Selectric Composer Practice Form	NEW
	549-0057-0	Selectric Composer Practice Form	NEW
	549-0058-0	Selectric Composer Practice Form	NEW
	549-0059-0	Selectric Composer Practice Form	NEW
	549-0060-0	Selectric Composer Practice Form	NEW
	549-0704-0	IBM Mag. Tape Sel. Comp. - Guide - Copy Prep.	NEW
	549-0705-0	IBM Mag. Tape Sel. Comp. - Record Unit Training Guide	NEW
	549-0707-0	IBM Mag. Tape Sel Comp - Output Unit Training Guide	NEW
	549-0710-0	Guide to Copy Prep. - Sample Projects	NEW
	549-0711-0	Layout Information Sheets	NEW
	549-0712-0	Prenumbered Paper 8 1/2 x 14	NEW
	549-0713-0	Prenumbered Paper 14 x 14	NEW
	A22-6823-4	IBM S/360 Input/Output Configurator	Scrap A22-6823-2, -3
	A22-6916-0	IBM S/360 Model 85 Functional Characteristics	NEW
	A22-6918-0	IBM 2420 Model 7 Magnetic Tape Unit Component Description Bulletin	NEW
	A22-6920-0	IBM S/360 Model 85 Configurator	NEW
	A24-1034-3	82, 83 & 84 Sorters	Use A24-1034-2
	A33-1500-0	IBM S/360 Model 20 Serial I/O Channel OEMI	NEW
	C24-1404-9	1401 Installation Manual -- Physical Planning	Use C24-1404-8, N24-0377 & N24-0407
	C24-3392-7	IBM S/360 Basic Programming Support: DASD Utility Programs Operating Guide	Use C24-3392-6, N21-5057 & N21-5062
	C24-3413-3	IBM S/360 Basic Programming Support, Sort/Merge Programs Operating Guide	Use C24-3413-2, N21-5046 & N28-2244
	C24-5024-3	IBM S/360 BOS Operator Messages	Scrap C24-5024-2, -1, N24-5116, N24-5155, N24-5168, N24-5187, N24-5269, N24-5326 & N24-5325
	C24-5060-2	BOS System Generation & Maintenance	Scrap C24-5060-0, N24-5261, N24-5315, N24-5332 & N24-5338
	C24-9004-1	IBM S/360 Model 20 I/O Control System for the Communications Adapter Operator Procedures	Use C24-9004-0, N24-9025 & N33-8510
	C28-6668-0	IBM S/360 Operating System: Sort/Merge Timing Estimates for IBM 2420 Model 7 Magnetic Tape Unit Planning Guide	NEW
	E20-0301-0	Plant Management System (PMS) for Electric Utilities	NEW
	H20-0149-2	IBM S/360 Online Teller Program and Mortgage Loan Background Capability (360A-FB-16X) Application Description	Scrap H20-0149-1
	J20-0030-0	QUIKTRAN TELETYPEWRITER Terminals Manual	NEW
	K20-0225-0	School Information Center in Iowa's Tri-County Area 9	NEW
	K20-0246-0	Control Center at Hough Division International Harvester Co.	NEW
	N20-1069	TNL to a Programmer's Introduction to the IBM S/360 Architecture, Instructions, and Assembler Language Re: C20-1646-4	NEW
	N20-1072	TNL for 1130 Type Composition Program (1130-DP-04X) Operator's Manual Re: H20-0288-1	NEW
	N20-1073	TNL for 1130 Type Composition Program (1130-DP-04X) Programmer's Manual Re: H20-0287-1 with N20-1056-0	NEW
	N22-0282	Changes to IBM S/360 Principles of Operation Re: A22-6821-2, -3, -4, -5, -6 with N22-0231 to -2 & -3, N22-0232 to -2 & -3, N22-0236 to -2, -3 & -4, N22-0250 to -2, -3 & -4, N22-0244 to -2, -3, -4 & -5, N22-0278 to -2, -3, -4, -5 and -6	NEW
	N22-0285	Changes to IBM S/360 System Summary Re: A22-6810-8 with N22-0276 & N22-0286	NEW
	N23-0614	ENL to IBM S/360 QTAM Learners Guide I Re: R20-1061-0	NEW
	N24-0413	IBM 1418 Optical Character Reader IBM 1428 Alphameric Optical Reader Re: A24-1473-2	NEW - Scrap N24-0383
	N24-5321	TNL to BOS CNT Prog. & Assem. Oper. Guide Re: C24-3450-2	NEW
	N24-5328	TNL to S/360 BOS Programmer's Guide Re: C24-3372-6 with N24-5336	NEW

PRL #5 (Continued)

N24-5341	TNL to S/360 BOS Assembler with I/O Macros Re: C24-3361-5 with N24-5314 & N24-5335	NEW
N27-1293	IBM S/360 Transitions Aids: FORTRAN II Language Conversion Program for the IBM 1401 Re: C28-6560-1 with N27-1241	NEW
N30-2506	TNL IBM S/360 Operating System Remote Job Entry Re: C30-2006-0	NEW
N33-1506	TNL to IBM S/360 Model 20 Functional Characteristics Re: A26-5647-2, -3 with N26-0174, N26-0179, N33-1504, N26-0176, N33-1501 & N33-1507	NEW
R20-1004-2	IBM 519 Doc. Orig. Mach. Education Guide	Use R20-1004-1
R29-0071-5	IBM Basic Computer System P.I. Volume 1	Use R29-0071-4
R29-0072-5	IBM Basic Computer Systems P.I. Volume 2	Use R29-0072-4
R29-0205-3	IBM S/360 COBOL - COBOL Fund. P.I. Text	Use R29-0205-2
V20-6061-0	Model 85 Desk Flipcharts	NEW
Y20-0155-0	Medical Information System Programs (MISP) (360A-UH-08L) Programmer's Manual	NEW
Y20-0158-0	TNL for 1130 Type Composition Program (1130-DP-04X) System Manual - Volume I Re: Y20-0040-1 with Y20-0138-0	NEW
Y20-0162-0	COBOL/Assembler Tape Read Error Recovery Routine Sales & Systems Guide	NEW
Y24-5012-1	IBM S/360 BPS Tape Assembler PLM	Scrap C24-3443-0, Z24-5012-0, Y24-5046 & Y24-5012-0
Y24-5076	TNL to S/360 BOS System Control PLM BSC & 1287 Support Re: Y24-5002-0 with Y24-5031, Y24-5044 & Y24-5061	NEW
Y24-9004-1	IBM S/360 Model 20 Punched-Card Utility Programs, PLM	Use Y24-9004-0, Y24-9008 & Y33-8500
Y28-3052-0	IBM S/360 Time Sharing System Supervisor PLM Re: Y28-2012-0	NEW
Y28-3053	IBM S/360 Time Sharing System Command Language PLM Re: Y28-2013-0	NEW
Z77-8002	Generating a Disk Operating System - Version 2	NEW
Z77-8007-0	New Methods of Cost Accounting in Banking	NEW
Z77-8008-0	Analysis of Balance Sheets	NEW
Z77-8009-0	Computation of Tooth Profiles for Gear Milling Cutters for Plot	NEW

February 23, 1968 Issue No. 68-03

IBM INSTALLATION NEWSLETTER
TABLE OF CONTENTS

OS/360 DEBUGGING AIDS

The OS/360 Programmer's Guide to Debugging (C28-6670-0) will be expanded in the future to include techniques to be used when debugging high level language programs. The following explanations are preliminary to that expansion for the use with COBOL programs' ABEND Dump. They are futher explanations to aid in determining the exact reasons for the abnormal termination of the program.

Completion Code #	Explanation or Problem and Possible Solutions
F13	Check register 2 of registers at the entry to ABEND. This address points to the DCB in conflict.
OC1	Interrupt at 000050. Look at register 1 of the registers at entry to ABEND. Add hex 30 to the reg 1 address. This should point to the DDname of a missing DDcard.
OC1	When the interrupt is at 00004A, look for no //SYSOUT DD SYSOUT=A card, any missing JCL card, or the wrong name of a JCL card. Add hex 30 to register 1 of the registers at entry to ABEND. This should point to the DDname of the error DDname.
OC1	When the interrupt is at 000484, look for writing an unopened output in COBOL program.
OC1	Interrupt at 00004F. Look for inconsistent JCL and external name in the COBOL program.
OC4	Check for the block size and record size being equal for variable record input or output.
OC4	Check for moving data to an unopened output.
OC4	If error occurs during COBOL compilation, check for missing select statement.
OC4	Interrupt at 004814, look for reading an unopened input in a COBOL program or a missing DDcard.
OC5	Check for duplicate close of an input or output file if DS formatting discontinued.
OC5	If a sort is being attempted, the SCRT is being attempted with an incorrect catalog procedure. Use catalog procedure 'COBFSORT' or 'COBTSORT'.
001	Register 1 of the SVRB points to the DCB which caused the I/O problem. Look for input record and blocking errors. That is, the input does not agree with the record and blocking descriptions in the DCB or COBOL file description.
002	Register 2 of registers at the entry to ABEND contains the address of the DCB for the file causing the I/O problem. Check the DCB list for the specific file.
213	Check register 14 of the registers at entry to ABEND. This address points to the file which has no DSCB.
400	If this ABEND code occurs during a compile step, the member to be compiled has not been extracted from the source library for compilation.

OS/360 RELEASE 14 USE OF 1600 BPI TAPE

When a data set is cataloged by DISP=CATLG, the device type code in the UCB is stored as the catalog entry. This code is used during allocation to match the SYSGEN'ed device name and device mask table entries.

With Release 14, the SYSGEN'ed generic names for magnetic tape have been broadened to include 2400 Model #4, 5, 6 with and without the dual density 800 BPI feature.

Data sets can be retrieved from the catalog according to:

1. Data sets cataloged with Release 13 or before as being on 2400 Model 4, 5, 6 with dual density can only be retrieved on dual density tapes unless the user re-catalogs using Release 14 to reflect the density of the data set.

2. Data sets cataloged with Release 14 as being on 2400 Model 4, 5, 6 with dual density will be cataloged according to their density.

   Therefore, an 800 BPI data set can be retrieved on 2,400 Model 1, 2, 3, or Model 4, 5, 6 with dual density, while a 1600 BPI data set can be retrieved on 2400 Model 4, 5, 6 with or without dual denisty.

3. Data sets cataloged with Release 13 or before as being on 2400 Model 1, 2, 3 can be retrieved on 2400 Model 1, 2, 3 or Model 4, 5, 6 with dual denisty.

4. Data sets cataloged with Release 13 or before as being on 2400 Model 4, 5, 6 without dual density can be retrieved on 2400 Model 4, 5, 6 with or without dual density.

In summary, data sets cataloged on dual density tapes before Release 14 can create an allocation problem and should be recataloged using Release 14 if an installation has mixed device types.

OS/360 SYSOUT LINE LIMIT CAPABILITY

Job steps which direct output to SYSOUT can have control over the number of lines put out through the space parameter. Combining symbolic parameters with the space subparameters creates an extremely flexibile and effective DD card line limit capability.

Example

```
// Step 1      EXEC   PGM=NONE, TRK=20
//OUTPUT       DD     SYSOUT=A, DCB=
                      (Recfm=FB, LRECL=
                      132, BLKSIZE=3625)
                      SPACE=(3625, (&TRK),
                      RLSE, , ROUND)
```

OS/360 COMBINED BISAM/QISAM FOR PERFORMANCE IMPROVEMENTS

The following is a contribution from an IBM Field Systems Center. Potential users should first evaluate its usefulness in their own environment.

OS/360 ISAM Applications requiring sequential reading and updating of a file can use QISAM for reading and do the updating of the file using BISAM. Two DCB's are required--one for BISAM and one for QISAM. Whenever a BISAM update is performed, it is necessary to post the following fields from the BISAM DCB to the QISAM DCB:

DCBLEMIZ, DCBLEM13 DCBNOREC, DCBRORGI, DCBRORG2, DCBNREC, DCBLPDA, DCBLECI, DCBLEMI1, DCBLIOV, DCBNBOV

When reorganizing, the following QISAM DCB fields must be posted to the BISAM DCB:

DCBTDC
DCBRORG3+1 (3 bytes)

The primary advantage of this method of file management is the possible increase in throughput.

OS/360 DATA CHAINING

A knowledge of OS/360 Data Chaining and its use is necessary when computing Channel Utilization and / or CPU interference.

Data Chaining is implemented by a number of OS/360 Program Components. However, most programs actually Data Chain using a count that is the physical record length. That is, the next CCW is selected by the channel during physical record length.

Examples of this are: (Not a complete list)

1. Scatter Fetch where each CSECT is at least one physical record.

2. Record overflow where the physical track boundaries cause a delay gap.

3. Dump-Restore where each physical record is a new CCW.

There are, however, two Program Components which do implement "Scatter Read/Gather Write". That is, data chaining logical segments within a physical record. The only cases known to be in OS/360 are:

1. Exchange Buffering where each logical segment of the physical record is a new CCW. Ref. SRL C28-6646.

2. Teleprocessing BTAM/QTAM when variable length messages require dynamic data chaining of storage buffers.

OS/360 ISAM RECORD COUNTING WITH IEBISAM

This article is an IBM field contribution and the method was tested by the contributor under OS/360 Release 13 on a S/360 Model 40 with Index Sequential files on a 2314 direct access storage device. Potential users should be aware that it has not been submitted to any formal IBM test, that no maintenance can be expected and should first evaluate its usefulness in their own environment.

The following method can be used to provide a count of logical records for ISAM files when the IEBISAM utility is used to:

1. Create backup tapes for Index Sequential files.

2. Reorganize Index Sequential files.

Exhibit 1 shows the necessary code to be applied to the source language of IEBISAM utility modules. This will provide logical record counting in both the "LOAD" and "UNLOAD" facilities of the utility. Logical record counts are written to SYSOUT and to the console typewriter at the end of each run.

It should be noted that a SYSPRINT DD statement must be provided when record counting is required. All other requirements and facilities of the IEBISAM Utility remain the same. (See SRL C28-6586-6).

OS/360 ISAM RECORD COUNTING WITH IEBISAM

```
*  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *
*
* THE FOLLOWING ADDITIONS AND REVISIONS TO IEBISAM SOURCE LANGUAGE WILL
* PROVIDE RECORD COUNTS FOR BOTH LOAD AND UNLOAD FUNCTIONS OF THE
* UTILITY - RECORD COUNT MESSAGES ARE WRITTEN TO SYSOUT AND THE CONSOLE
* TYPEWRITER.       SOURCE LANGUAGE AS OF OS/REL 13.
*                                                                       *
*  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *
*                                                                       *
**TO IEBISR - F01JAN67 ADD THE FOLLOWING CARD IN SEQUENCE
*                                                                       *
ISCOUNT  DS    F               RECORD COUNT                             01590000
*                                                                       *
*  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *
*                                                                       *
**TO IEBISR - F01JUN67 REVISE THE FOLLOWING CARD FROM - TO
*                                                                       *
         BE    PRTCOUNT                NEW CARD  INSERT                 01360000
         BE    CONTROL                 OLD CARD  REMOVE                 01360000
*                                                                       *
*  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *
*                                                                       *
**    IEBISR            ADD THE FOLLOWING CARDS IN SEQUENCE
*                                                                       *
RELEV    EQU   11                  RECORD COUNT WORK REGISTER           00910000
GOGOX    MVC   GOGO+40(8),MESS8+32 MOVE COUNT TO CONSOLE MESS           01410000
GOGO     WTO   'NUMBER OF RECORDS TRANSMITTED -        '                01411000
PRTCOUNT L     RELEV,36(WORKREG)   GET COUNT FROM COMMUN AREA           02044000
         CVD   RELEV,DWCOUNT                                            02045000
         UNPK  MESS8+32(8),DWCOUNT(8)                                   02046000
         OI    MESS8+39,X'F0'      ELIMINATE SIGN                       02047000
         MVI   31(WORKREG),X'20'                                        02048000
         B     GOGOX                                                    02049000
         DC    X'00'                                                    02690000
         DC    AL3(MESS8)                                               02691000
MESS8    DC    CL32'NUMBER OF RECORDS TRANSMITTED - '                   03410000
         DC    CL32'                               '                    03411000
         DC    CL32'                               '                    03412000
         DC    CL34'                                 '                  03413000
*                                                                       *
*  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *
*                                                                       *
**TO IEBISL ADD THE FOLLOWING CARDS IN SEQUENCE. LEVEL F01JUN67
*                                                                       *
WK7      EQU   7                   RECORD COUNT - LOAD                  00950000
         ST    WK7,36(WK4)         PASS COUNT TO COMMON AREA            01910000
         SR    WK7,WK7             CLEAR RECORD COUNT                   02370000
         BCT   WK7,*+4             DECREMENT COUNT AGAINST INIT FALSE COUNT 02770000
         LA    WK7,1(WK7)          INCREMENT RECORD COUNT               02950000
*                                                                       *
*  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *
*                                                                       *
**TO IEBISU ADD THE FOLLOWING CARDS IN SEQUENCE. LEVEL F01JAN67
*                                                                       *
WK7      EQU   7                   RECORD COUNT - UNLOAD                01030000
         SR    WK7,WK7             CLEAR RECORD COUNT                   01390000
         ST    WK7,36(WK4)         PASS COUNT TO COMMON AREA            02210000
         LA    WK7,1(WK7)          INCREMENT RECORD COUNT.              02710000
*                                                                       *
*  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *
```

Exhibit 1

DOS/360 REDUCING SIZE OF CORE IMAGE LIBRARY

The following contribution has not been submitted to any formal IBM test. It has been tested by the author. Potential users should first evaluate its usefulness in their own environment.

The following write-up describes a procedure for reducing the size of the C. I. L. (Core Image Library). It involves independently assembling and cataloging each of the logic modules used in an installation into the C. I. L. rather than into the R. L. By using this method, a logic module will appear in the C. I. L. only once, rather than repeatedly in the C. I. L. with every program using that particular type of file. When a processing program requires the logic module, the programmer will code (into his program) the few statements needed to "LOAD" the module into core at execution time.

The trade-off for saving space in the library is that after the first phase of a processing program has been loaded into core by Job Control, one additional access will have to be made to the C. I. L. for every logic module needed in the program. Considering the total running time of any given program, however, these accesses will usually be negligible, especially since the reduced size of the library should also reduce the average seek time.

One other consideration is that the Autolink feature will no longer be applicable; and therefore the systems programmer will have to give more thought to the assembling of the modules in regard to subsets and supersets, and will also have to provide exact documentation for other programmers on the modules and their loading procedure.

(Although the author has not tested it, she believes that the I/O areas could be reserved in the common area in the same way).

1. Job "ONE" (Below) shows how to assemble and catalog a logic module into the C. I. L. This procedure would be repeated for each logic module commonly used in the installation, assigning a different name to installation, assigning a different name to each in the PHASE card, and making each self-relocating.

```
// JOB ONE
// OPTION CATAL
 PHASE MOD1,+0
// EXEC ASSEMBLY
        START 0
        PRMOD

/*
// EXEC LNKEDT
/&
```

2. Job "TWO" (Exhibit 1) shows (encircled) the statements needed to define an area in the problem program into which the PRMOD will be loaded, the DTF entry required, and the additional statements needed to LOAD the module and to improve linkage to the module from the DTF.

The statements on the line following the encircled statements show how to LOAD in a second module. The procedure would be the same for a third, etc.

To determine the size of the DS area for each module in the Com area, use the listing from the assembly of each module. Be certain to define the modules on double word areas or a specification error will occur.

DOS/360 REDUCING SIZE OF CORE IMAGE LIBRARY

IBM IBM System/360 Assembler Coding Form

PROGRAM | PUNCHING INSTRUCTIONS | GRAPHIC | PUNCH | PAGE OF | CARD ELECTRO NUMBER

PROGRAMMER | DATE

STATEMENT

Name	Operation	Operand	Comments	Identification-Sequence
// JOB TWO				
// OPTION LINK				
// EXEC ASSEMBLY				
PROG	START			
BEGIN	BALR			
	USING			
	MVC	PRINT+17(3),=AL3(ABC)	PUT ADDR. OF LOGIC MOD IN DTF. THE V-TYPE ADDR. CONSTANT WITH THE NAME OF THE LOGIC MOD. (ABC), WILL NOT BE RESOLVED. YOU CANNOT USE AN "ENTRY" STMT. IN A COMMON. SO EXPECT AN UNRESOLVED EXTRN AT LINK EDIT FOR EACH MODULE. IF IT IS POSSIBLE THAT A MOD. MIGHT BE IN THE R.L. CALLED "ABC", USE NOAUTO FOR THIS PHASE.	
	MVC	PUNCH+17(3),=AL3(XYZ)		
	OPEN	PRINT,PUNCH		
	LOAD	MOD1,ABC		
	LOAD	MOD2,XYZ		
	PUT	PRINT		
	PUT	PUNCH		
EOJ	CLOSE	PRINT,PUNCH		
PRINT	DTFPR	———— ,MODNAME=ABC		
PUNCH	DTFCD	,MODNAME=XYZ		
	COM			
ABC	DS	4D	USE COM TO RESERVE SPACE FOR LOGIC MODULE IN CORE WITHOUT TAKING UP SPACE IN THE C.I.L. BE SURE THAT "DS" WITHIN COM IS ON A DOUBLE WORD FOR PROPER ALIGNMENT.	
XYZ	DS	22D		
	END	BEGIN		
/*				

Exhibit 1

DOS/360 PROGRAMMING DISK WORK FILES

The following is a field contribution which has not been submitted to any formal IBM test. Potential users should first evaluate its usefulness in their own invoronment.

The purpose in using work files is that it is possible to add records to sequential files without reading through the existing records on the file. Defining the file as a work file (rather than input or output) makes it possible to access any record on the file. Saving the address of the last record written on a file makes it possible to start the next run at that point.

The programmer needs to perform several manipulations upon opening and closing a work file. DTFSD work files does not support blocked records. For these two reasons, a detailed description might be helpful to others programming this type of file. Exhibit 1 shows the coding and logic.

1. Storing Transitory Information

   a. An area at the beginning of the file should be reserved for storing information, such as the address of the last record written, etc. The choice was made to leave the entire first track vacant, so that when sorting the disk, one could start at the second track.

   b. The file should be opened as OUTPUT. A method is needed to dtermine if loading is to start at the beginning of the file, or is to continue at the location where the previous run ended. The choice was to use the UPSI Byte to determine this.

   c. When the last record has been processed, (but the last block has not been written), a NOTE (filename) should be issued. The disk address of the previous write is returned in GP Register 1 in the form 0CHR. The length remaining on the current track is returned in GP Register 0 in the form 00LL. The programmer should move this information (address and length) to a save area.

   d. Before writing the last block, the programmer should fill the remaining portion of the block with some special characters. This will be needed when re-using the file to dtermine where in the block processing is to resume. The choice was to fill out the last block with High-Value (HEXIDECIMAL FF).

   Now, the last block should be written, and then another NOTE (filename) should be issued. The disk address of the last block will be returned in GP Register 1 in the form 0CHR. The length remaining on the track will be returned in Register 0 in the form 00LL. The programmer should also move this information to a save area. Now, a CLOSE (filename) should be issued. This puts an EOF at the beginning of the next block.

   Next, the information saved from the two previous notes should be moved to the OUTPUT area, so that it may be written on the disk. The first six bytes of OUTPUT area was chosen to store information from the second note; and the second six bytes to store information from the first note.

   Next, an OPEN (filename) should be issued, followed by a POINTS (filename), then a WRITE (filename) UPDATE (area), and a CLOSE to the file.

2. Starting At a Point Where The Previous Run Ended Adding Records

After opening, the programmer must determine at what (CHR) Location the writing of records is to begin. The information stored from the previous run should be obtained from the first 12 bytes on the disk file. This was accomplished by issuing a Point S to the file, then a READ (filename) SQ (area). The information from the first and second notes should be moved to separate storage areas (six bytes each). Next, a POINT R to the last record written on previous run (using second note information) is issued. This is followed by READ (filename) SQ (area). Then, the next available logical record in this particular block is located, followed by a return to normal processing, except that a POINT W (using first name information) prior to the first write should be provided for.

3. Starting At the Beginning - New File

   After opening, merely write enough dummy blocks to fill the first track before acutal processing is begun.

2. In Exhibit 2, A and B should be defined explicitly as datanames. Additionally, an A is X'C1', not X'01' and B is X'C2', not 'X02'.

DOS/360 COBOL ISFMS DATA ON 2321 INDICES ON 2311 CORRECTION

An article on the above subject was published in Newsletter 68-02, page 10. The following is a clarification and correction to that article.

1. A change was made with the latest Release (15) of DOS/360 to add the Master Index on the 2321 by moving X'02' into byte 21 of the skeleton DTF and X'F2' for the 2311. It is not necessary to make the move into byte 200 or byte 65 as stated in the article. See page 109 of the DOS/360 Programmers Guide.

DOS/360 PROGRAMMING DISK WORK FILES

```
C NO....5....1....5....2....5....3....5....4....5....5....5....6....5....7....5....8
  1 **
  2 NOP1      NOP   MAINLINE  OPEN SWITCH
  3           OPEN  FILENAME
  4           MVI   NOP1+1,X'F0' ALTER OPEN SWITCH TO PROCEED TO MAINLINE.
  5 *
  6 ** DETERMINE IF ADDITION TO EXISTING FILE  AND BRANCH ACCORDINGLY.
  7 *
  8 ** IF STARTING AT BEGINNING OF FILE ISSUE WRITES TO FILL FIRST TRACK.
  9 *
 10 ** IF ADDING RECORDS TO AN EXISTING FILE . . .
 11 *
 12           POINTS FILENAME
 13           READ   FILENAME,SQ,I/O AREA  READS FIRST RECORD ON FILE.
 14           CHECK  FILENAME
 15           MVC    SAVE1(6),I/O AREA  SAVE NOTE 1
 16           MVC    SAVE2(6),I/O AREA+6  SAVE NOTE 2
 17           LA     2,SAVE2
 18           POINTR FILENAME,(2)
 19           READ   FILENAME,SQ,I/O AREA  READS LAST BLOCK WRITTEN PREVIOUSLY
 20           CHECK  FILENAME
 21           MVI    NOP2+1,X'F0' ALTER 1ST SWITCH TO BRANCH TO REWRITE ROUTINE.
 22 *
 23 ** FIND NEXT AVAILABLE LOGICAL RECORD WITHIN BLOCK AND GO TO MAINLINE.
 24 *
 25 MAINLINE  ADD LOGICAL RECORDS UNTIL BLOCK IS FILLED,THEN GO TO WRITE ROUTINE.
 26 *
 27 WRITE     CHECK  FILENAME
 28 NOP2      NOP    REWRITE   1ST WRITE SWITCH
 29           WRITE  FILENAME,SQ,I/O AREA
 30 ** GO BACK TO MAINLINE
 31 *
 32 ** TO WRITE BACK THE FIRST BLOCK WHEN ADDING TO EXISTING FILE . . .
 33 *
 34 REWRITE   LA     2,SAVE1
 35           POINTW FILENAME,(2)
 36           WRITE  FILENAME,SQ,I/O AREA  THIS REWRITES BLOCK
 37           MVI    NOP2+1,X'00' CLOSES FIRST WRITE SWITCH.
 38           MVI    QUIT+1,X'00'   INDICATES IF A BLOCK HAS BEEN WRITTEN.
 39           MVI    NOP3+1,X'00'   INDICATES IF A BLOCK HAS BEEN WRITTEN.
 40 ** THE 2 ABOVE INSTRUCTIONS ALTERS QUIT ROUTINE TO BYPASS SAVING NEW ADDRESS
 41 ** BECAUSE THE PREVIOUSLY SAVED ADDRESSES ARE STILL VALID.
 42 ** GO TO MAINLINE.
 43 ** TO END THIS JOB . . .
 44 QUIT      B      SPLPNTW  BRANCH IF NO BLOCKS WERE WRITTEN DURING THIS RUN.
 45 *
 46 ** OTHERWISE SAVE NEW ADDRESS FOR FUTURE RUNS . . .
 47           CHECK  FILENAME
 48           NOTE   FILENAME
 49           ST     1,SAVE1
 50           STH    0,SAVE1+4
 51 ** FILL OUT REMAINING PORTION OF BLOCK WITH SPECIAL CHARACTERS.
 52           WRITE FILENAME,SQ,I/O AREA
 53           CHECK  FILENAME
 54 NOP3      B      CLOSEOUT BYPASSES 2ND NOTE IF THE ABOVE WRITE
 55 *                       IS THE ONLY WRITE ISSUED DURING THIS RUN.
RC NO....5....1....5....2....5....3....5....4....5....5....5....6....5....7....5....8
```

Exhibit 1

DOS/360 PROGRAMMING DISK WORK FILES

```
RC NO....5....1....5....2....5....3....5....4....5....5....5....6....5....7....5....8
   56 ** OTHERWISE SAVE NEW ADDRESS FOR FUTURE RUNS . . .
   57           NOTE  FILENAME
   58           ST    1,SAVE2
   59           STH   0,SAVE2+4
   60 *
   61 * MOVE SAVE1 & SAVE2 TO I/O AREA.
   62           CLOSE FILENAME  PUTS EOF AT BEGINNING OF NEXT BLOCK.
   63 *
   64 ** TO RECORD INFORMATION FROM NOTE1 AND NOTE2 . . .
   65           OPEN  FILENAME
   66           POINTS FILENAME
   67           CHECK  FILENAME
   68           WRITE FILENAME,UPDATE,I/O AREA-
   69           CHECK  FILENAME
   70 CLOSEOUT CLOSE FILENAME
   71 *         STOP
   72 SPLPNTW   LA    2,SAVE1
   73           POINTW FILENAME,(2)
   74 *
   75 ** GO TO ** FILL OUT REMAINING PORTION OF BLOCK.

RC NO....5....1....5....2....5....3....5....4....5....5....5....6....5....7....5....8
```

Exhibit 1

DOS/360 PROTECTION OF COBOL ISFMS FILES

The following field contribution has not been submitted to any formal IBM test. Potential users should first evaluate its usefulness in their own environment.

When a program check interruption occurs in a DOS COBOL program, the normal action is for the job to be immediately cancelled. If the job is one that has been adding records to an index sequential master file, then this immediate cancellation would make the file unusable because the file would not have been closed and, therefore, data pertinent to the file organization would not have been posted in the file labels.

A solution to this problem is to call an Assembler subroutine from the COBOL program that will execute the STXIT macro instruction. The STXIT macro provides the facility to establish linkage from the Supervisor to a user's routine for Program Check Interrupt Handling. The user can utilize this routine to close his index sequential file and take any other action he desires when a program check occurs.

Being able to close the file saves the user from having to recreate his ISFMS master file.

Exhibit 1 shows a sample COBOL program which calls an Assembler subroutine (Exhibit 2) to issue the STXIT macro. Notice that in Statement 13 of Exhibit 2 Register 14 (the return register) has been incremented by a value of 8. This is to allow for bypassing the first 8 bytes of the COBOL return coding. The reason for this is that COBOL clears all user program check routines on normal returns from a call (only when you have arithmetic statements; otherwise, it does not).

DOS/360 PROTECTION OF COBOL ISFMS FILES

```
LINE NO.  SEQ. NO.        SOURCE STATEMENT

      1          IDENTIFICATION DIVISION.
      2          PROGRAM-ID. @BLDEMP@.
      3          DATE-WRITTEN. OCTOBER 17,1967.
      4          ENVIRONMENT DIVISION.
      5          CONFIGURATION SECTION.
      6          SOURCE-COMPUTER. IBM-360 F30.
      7          OBJECT-COMPUTER. IBM-360 F30.
      8          INPUT-OUTPUT SECTION.
      9          FILE-CONTROL.
     10              SELECT EMP-CTL ASSIGN TO @SYS010@ DIRECT-ACCESS 2o1
     11              ACCESS IS RANDOM
     12              ORGANIZATION IS INDEXED
     13              SYMBOLIC KEY IS SYM-KEY
     14              RECORD KEY IS REC-KEY.
     15              SELECT INPUT-FLE ASSIGN TO @SYS011@ UNIT-RECORD 254
     16          DATA DIVISION.
     17          FILE SECTION.
     18          FD  EMP-CTL,
     19              BLOCK CONTAINS 20 RECORDS,
     20              RECORDING MODE IS F,
     21              LABEL RECORDS ARE STANDARD,
     22              DATA RECORD IS EMP-CTL-R.
     23              01 EMP-CTL-R.
     24                 02 EMP-NME PICTURE X%22¤.
     25                 02 REC-KEY.
     26                    03 EMP-NO PICTURE X%5¤.
     27                    03 P-EMP-NO PICTURE X%2¤.
     28                    03 SUPM-CD     PICTURE X%1¤.
     29                    03 YR    PICTURE X%2¤.
     30                    03 MO     PICTURE X%2¤.
     31                 02 IND-NO  PICTURE  X%3¤.
     32                 02 BATCH-NO  PICTURE X%4¤.
     33                 02 TAX-WAG PICTURE X%10¤, JUSTIFIED RIGHT.
     34                 02 EMP-CT  PICTURE X%5¤.
     35          FD  INPUT-FLE, RECORDING MODE IS F, LABEL RECORDS
     36              ARE OMITTED, DATA RECORD IS INPUT-REC.
     37              01 INPUT-REC.
     38                 02 IBATCH-NO  PICTURE X%4¤.
     39                 02 FILLER     PICTURE X%1¤.
     40                 02 ISUPM-CD     PICTURE X%1¤.
     41                 02 FILLER     PICTURE X%3¤.
     42                 02 IMO     PICTURE X%2¤.
     43                 02 IYR     PICTURE X%2¤.
     44                 02 FILLER      PICTURE X%3¤.
     45                 02 IP-EMP-NO  PICTURE X%2¤.
     46              02 FILLER      PICTURE X.
     47                 02 IEMP-NO     PICTURE X%5¤.
     48                 02 IEMP-NME     PICTURE X%22¤.
     49                 02 FILLER      PICTURE X%12¤.
     50          02  ITAX-WAG PICTURE 9%9¤.
     51              02 FILLER  PICTURE X%5¤.
     52              02 IIND-NO  PICTURE X%3¤.
     53              02 FILLER   PICTURE X%5¤.
     54          WORKING-STORAGE SECTION.
```

Exhibit 1

DOS/360 PROTECTION OF COBOL ISFMS FILES

```
LINE NO.  SEQ. NO.        SOURCE STATEMENT
      55          77 SYM-KEY     PICTURE X%12¤.
      56          77  TWAG PICTURE IS 9999 VALUE IS ZEROES.
      57          PROCEDURE DIVISION.
      58              ENTER LINKAGE.
      59              CALL @STXITPC@.
      60              ENTER COBOL.
      61          FST.  OPEN OUTPUT EMP-CTL INPUT INPUT-FLE.
      62          BEGIN. READ INPUT-FLE AT END GO TO END-RTN.
      63                  MOVE IEMP-NME TO EMP-NME.
      64                  MOVE IP-EMP-NO TO P-EMP-NO.
      65                  MOVE IEMP-NO TO EMP-NO.
      66                  MOVE IYR TO YR.
      67                  MOVE IMO TO MO.
      68                  MOVE IIND-NO TO IND-NO.
      69                  MOVE IBATCH-NO TO BATCH-NO.
      70                  MOVE ITAX-WAG TO TAX-WAG.
      71                  MOVE SPACES TO EMP-CT.
      72                  MOVE REC-KEY TO SYM-KEY.
      73                  MOVE ISUPM-CD TO SUPM-CD.
      74              ADD ITAX-WAG TO TWAG.
      75          WTE.    WRITE EMP-CTL-R.
      76                  GO TO BEGIN.
      77          END-RTN. CLOSE INPUT-FLE, EMP-CTL.
      78              STOP RUN.
      79              ENTER LINKAGE.
      80              ENTRY @CLOSE@.
      81              ENTER COBOL.
      82              CLOSE EMP-CTL INPUT-FLE.
      83              DISPLAY @FILES ARE CLOSED@ UPON CONSOLE.
      84              STOP RUN.
```

Exhibit 1

DOS/360 PROTECTION OF COBOL ISFMS FILES

```
STMT    SOURCE STATEMENT                                        DOS CL2-4 12/22.
   1 STXITPC  START 0
   2          USING *,15              COBOL LOADS REG 15
   3          SAVE  %14,12¤           SAVE REGISTERS
  4+* 360N-CL-453  SAVE       CHANGE LEVEL  2-0
  5+          STM   14,12,12+4*%14+2-%14+2¤/16*16¤%13¤
   6          LA    1,SAVEAREA        AREA FOR SUPVR TO STORE PSW & REGISTERS
   7          LA    0,PRGCHK          ADDR OF USERS PROGRAM CHECK RTN
   8          STXIT PC,%0¤,%1¤
  9+* 360N-CL-453  STXIT      CHANGE LEVEL  2-0
 10+          DC    0H@0@
 11+          SVC   16
 12           LM    14,12,12+4*%14+2-%14+2¤/16*16¤%13¤ RESTORE REGISTERS
 13           A     14,K8             BYPASS COBOLS CLEARING OF PROG CHK RTN
 14           BR    14                RETURN TO COBOL PROGRAM
 15 SAVEAREA DC    9D@0@             SUPVR WILL STORE PSW & REG 0-15 HERE
 16          DS    0F
```

Exhibit 2

DOS/360 PROTECTION OF COBOL ISFMS FILES

```
17 K8        DC    F@8@
18 PRGCHK    BALR  2,0          USERS PROGRAM CHECK RTN
19           USING *,2
20           L     15,#V%PCCLOSE¤ ADDR OF COBOL PROGRAM RTN TO CLOSE FILES
21           LA    13,CBLSAVE   ADDR OF AREA FOR COBOL TO SAVE REGS
22           BR    15           GO TO COBOL RTN TO CLOSE FILES
23 CBLSAVE   DC    9D@0@        AREA FOR COBOL PROG TO SAVE REGS
24           END
25                #V%PCCLOSE¤
```

Exhibit 2 (Continued)

DOS/360 COBOL USE OF DISPLAY TO SYSLST CLARIFICATION

The use of a DISPLAY to SYSLST in a label handling declarative may force an OPEN on SYSLST.

When the user is handling his own labels in a declarative section, the system will transfer control from the transient OPEN routine to the user's label routine. IF SYSLST is not OPEN and a display to SYSLST is issued in the label routine, an OPEN will be attempted. As an OPEN routine is already locked into the transient area, the OPEN for SYSLST will be invalid. A dummy display to SYSLST will have to be issued prior to the OPEN statement, using the declarative section. This subject will be clarified by a TNL to the Specification Manual.

DOS/360 PL/I-CARRIAGE CONTROL FOR RECORD I/O

The following technique, from an IBM Field Systems Center, has not been submitted to any formal IBM test. Potential users should first evaluate its usefulness in their own environment.

The attached listings give a reasonably convenient method for first character forms control. The theory behind this technique is to replace the DTF-A and DTF-T that were generated by PL/I. This is easily accomplished by assembling those tables separately and placing them on SYSLNK in front of the PL/I program to be link-edited. Also, there is a listing of an Assembler subroutine for testing for the occurrence of print overflow.

Exhibit 1 shows the assembly of DTF-A, DTF-T, and buffers for the print file. Because option LINK was set, the output of the assembly will be placed on SYSLNK and will be the first control section input to the linkage editor. Exhibit 2 illustrates the Assembler subroutine for print overflow. This program uses the standard linkage set-up discussed in DOS PL/I Programmer's Guide, page 37. Exhibit 3 shows the PL/I program. The file name in the PL/I program must agree with the control section name of the DTF-A assembly.

1130 FORTRAN ERROR CODE CORRECTION

The following is a correction to one of the FORTRAN Error Codes of the 1130 Disk Monitor, as listed in Table 9 of C26-3717.

Error Number C00. Program length exceeds capacity. The source program and the symbol table will not fit into core; thus, the source program must be shortened or the number of statement numbers decreased or both.

DOS/360 PL/I-CARRIAGE CONTROL FOR RECORD I/O

```
  SET DATE=01/03/68,CLOCK=00/00/00
  ASSGN SYSIN,X'00C'
ASSGN SYSLST,X'00E'
ASSGN SYSPCH,X'00D'
// JOB ASA
// OPTION LINK,NODECK,SYM,60C,  LISTX
// EXEC ASSEMBLY
FILEX    START
         DC    X'A2'
         DC    AL3(FILEY)
         DC    X'09'
         DC    AL3(0)
         DC    X'45'
         DC    X'00'
         DC    AL2(132)
         DC    AL4(0)
         DC    AL4(AREA1)
         DC    AL2(132)
         DC    AL2(0)
         DC    AL2(132)
FILEY    DTFPR IOAREA1=AREA1,IOAREA2=AREA2,CTLCHR=ASA,DEVADDR=SYSLST,  X
               BLKSIZE=132,DEVICE=1403,IOREG=(2),PRINTOV=YES
AREA1    DS    CL132
AREA2    DS    CL132
         ENTRY FILEY
         END
/*
```

Exhibit 1

DOS/360 PL/I-CARRIAGE CONTROL FOR RECORD I/O

```
// EXEC ASSEMBLY
OVRFLO   START
         USING *,9
         SAVE  (14,12)
         LR    9,15
         L     7,0(1)
         LA    1,PBL
         EXTRN FILEY,IJKSZCN
         CALL  IJKSZCN
         PRTOV FILEY,12,SKIP
         MVI   0(7),C'0'
         B     RETURN
SKIP     MVI   0(7),C'1'
RETURN   L     13,4(13)
         RETURN (14,12)
         DS    0F
PBL      DC    X'03'
         DC    AL3(INDIC)
         DC    A(80)
INDIC    DC    4X'0'
         END
/*
```

Exhibit 2

DOS/360 PL/I-CARRIAGE CONTROL FOR RECORDI/O

```
// EXEC PL/I
ASA:    PROCEDURE OPTIONS (MAIN);
        DECLARE
        FILEA FILE INPUT RECORD
          ENVIRONMENT (F(80) MEDIUM (SYSIPT,2540) BUFFERS(2));
        DECLARE
        FILEX FILE OUTPUT RECORD
          ENVIRONMENT (F(132) MEDIUM (SYSLST,1403) BUFFERS(2));
        DECLARE
        INPUT CHARACTER(80),
        1 OUTPUT,
            2 OUT2 CHARACTER (80),
            2 FILLER CHARACTER (52);
        DECLARE B CHARACTER (1) EXTERNAL;  /* PARAMETER FOR OVERFLOW */
        OPEN FILE (FILEA), FILE (FILEX);
        ON ENDFILE (FILEA) GO TO END;
        B='0';
READ:   READ FILE (FILEA) INTO (INPUT);
/*  FIRST COLUMN OF INPUT CARD USED FOR ASA CARRAGE CONTROL */
        OUT2=INPUT;
        WRITE FILE (FILEX) FROM (OUTPUT);
        CALL OVRFLO (B);                    /* TEST FOR OVERFLOW */
        IF B='1' THEN GO TO HEAD;
        GO TO READ;
HEAD:   OUT2='  SUCCESSFUL SKIP O K ';
        WRITE FILE (FILEX) FROM (OUTPUT);
        GO TO READ;
END:    END;
/*
// EXEC LNKEDT
// EXEC
 SPACE 1
OSPACE 2
-SPACE 3
1SKIP  1
```

Exhibit 3

NEW TYPE III PROGRAMS *

The following are the abstracts of programs which have been recently made available from the Type III library.

The program, along with its complete abstract, will be incorporated in subsequent issues of the Catalog of Programs.

Programs may be obtained by submitting a properly completed "General Program Request Card" (Form Number 120-1145-1) to the Program Information Department, 40 Saw Mill River Road, Hawthorne, New York, 10532.

These programs and their related documentation are distributed by IBM in the author's original form and have not been subjected to any formal testing.

Any discussion of Type III Programs must emphasize the following points:

1. **Type III programs are not part of the IBM product line, as are Programming Systems (Type I) or Application Programs (Type II).**
2. **Type III programs have not been subjected to any formal product test.**
3. **Recipients of Type III programs are expected to make the final evaluation as to the usefulness of the programs in their own environment.**
4. **There is no committed maintenance for Type III programs. However, any changes the author chooses to make will be announced in subsequent issues of the Catalog of Programs.**

* **NOTE: THE CUSTOMER MUST BE INFORMED THAT THE ABOVE APPLIES TO ANY OF THE FOLLOWING TYPE III ABSTRACTS FURNISHED TO HIM.**

S/360 A SIMULATION, EVALUATION AND ANALYSIS LANGUAGE. The Simulation, Evaluation and Analysis Language (SEAL) program consists of a compiler and an execution time package which runs under OS/360 with one private disk pack and 215 K bytes of core storage exclusive of OS requirements. SEAL is a high level discrete digital simulation language which includes the analytical capability of FORTRAN IV-G and a report generator to facilitate evaluation of results.

SEAL features are: automatic simulated time accounting, dynamic memory allocation, list manipulation, structured and dimensioned list members, dimension free structured arrays, and operations on character strings, bit strings, integer, real, complex and logical variables. SEAL also includes compile and object execution time source language diagnostics, ad hoc communication with the model, dump, trace and model maintenance facilities. No knowledge of JCL is required to run standard jobs or to use any of the above facilities.

SEAL is implemented in SEAL, assembly language and a few FORTRAN routines.

Ordering Procedure: Order File Number 360D-15.1.005.

The program material is available on a 9 track DTR only.

No tape submittal is required. The DTR will be provided by the Library.

S/360 SPEARMAN RANK-CORRELATION COEFFICIENT COMPUTATION S/360. The program computes the correlation between the rankings on two attributes within a sample comprising N observations. The attributes being ranked might be test (or course) grades in two subjects, or aptitude and achievement scores, etc. (In these examples, each student comprises an observation.) Ranks for each attribute as well as the correlation are computed by the program, according to the Spearman convention. The program was written in BPS FORTRAN and will execute on a minimum Model 30 or larger System/360. The source program is compatible with DOS FORTRAN and 1130 FORTRAN with appropriate changes of I/O unit logical addresses. As dimensioned in the source program, up to 99 observations are accepted, with up to three-digit raw scores for each of the attributes. In this form, the object program occupies 4034 bytes including data storage. Input is in card form, with a control card stating the value of N, followed by the raw scores for both attributes at one observation per card. Multiple samples may be stacked. Output is printed on-line. The program provides the practicing statistician the computational convenience of applying one of the standard non-parametric test of his discipline.

Ordering Procedure: Order File Number - 360D-13.3.001.

The program material can be obtained in card form or on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

The requester must indicate whether a 9 track or 7 track DTR is required. If not specified, a 9 track DTR will be provided.

No tape submittal is required. The DTR will be provided by the Library.

S/360 INTERSECTION DETECTION IN THREE DIMENSIONS: A TOOL FOR COMPUTER AIDED ENGINEERING DESIGN AND GRAPHIC DISPLAY. The intersection Detection Program (ID/3D) is a system of FORTRAN subroutines that enables the user to: 1) define 3-dimensional convex objects bounded by planes and quadric surfaces; 2) define line segments in 3-space; 3) test for intersections between pairs of objects; 4) test for intersections between segments and objects. The program is primarily a tool for the solution of pipe routing and component placement problems. The segment-object intersection test can also be used to solve the hidden line problem in computing graphic displays of 3-dimensional objects.

The program can be run under BPS or OS, and requires 128K core storage. The amount of core needed for compilation and linkage editing depends on the version of the compiler and linkage editor being used. The program has been compiled and tested under BPS with 128K storage, and under OS FORTRAN G and FORTRAN H with 512K storage.

Ordering Procedure: Order File Number 360D-08.7.004.

The program material can be obtained on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

The requester must indicate whether a 9 track or 7 track DTR will be provided.

No tape submittal is required. The DTR will be provided by the Library.

S/360 A CLUSTER ANALYSIS AND TAXONOMY SYSTEM FOR GROUPING AND CLASSIFYING DATA . Attempts to find the optimal way of organizing a set of objects into groups, according to one of several criteria. Each object is represented by a set of observations. Each criterion is a function of a partition of the set, and is intended to be a measure of the structure of that partition. Two measures are based on similarity coefficients computed between each pair of objects, and depend on a parameter running from 0 to 1. A tree structure can be created by running at several levels of this parameter. Other measures use variance-covariance criteria. The program searches for the optimal partition via a hill-climbing routine, which sifts through the partitions of the set, and which is designed to be effective not only for the encoded criteria, but for any criterion which can be formulated as a function of the partions of a set. For similarity based measures, the program will accept mixed discrete, rank-ordered and continuous data (decimal or integer) in a general input format and scales all input. Written G level FORTRAN and 360 assembly language and designed to run on the IBM/ 360 under OS. Three versions exist, for 128K , 256K and 512K machines.

Ordering Procedure: Order File Number
360D-06.7.005 (128K Version)
360D-06.7.006 (256K Version)
360D-06.7.007 (512K Version)

Tne program material can be obtained on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

The requester must indicate whether a 9 track or 7track DTR is required. If not specified, a 9 track DTR will be provided.

No tape submittal is required. Tne DTR will be provided by the Library.

S/360 SYSTEMS CARD ABSOLUTE LOADER PROGRAM (SCALP SYSTEM/360 MODEL 20 TAPE PROGRAMMING SYSTEM. SCALP is the system loader routine extracted from the Model 20 TPS Card Basic Monitor (360U-CL-137) which has been modified to run as a user program under the TPS tape resident system. SCALP permits the user to load RPG and BAL object decks from SYSRDR (2501) without disrupting a tape resident job stream.

When called into core from core image library, SCALP will perform loader functions for TXT, XRF, END, and REP cards and, unless overlayed, will remain core resident to execute FETCH for next phase deck. EOJ will overlay SCALP with SYSEOJ-job control. Written in BAL, SCALP loads at hex 680 and extends to 7D3.

Minimum machine configuration is that of TPS but present version of SCALP restricts SYSRDR to a 2501, Models AL or A2. Interrupts are cleared to allow double I/O areas for 2501-A2.

Ordering Procedure: Order File Number 360D-01.1.003.

Distribution will be in card form only.

JOB REWIND FOR S/360 MODEL 20 TO REWIND AND OPTIONALLY UNLOAD SPECIFIED LOGICAL TAPES. This program will greatly enhance the stacked job environment of the Tape/Disk Model 20. Job Rewind will rewind specified logical tape drives and optionally unload them. Device end is not significant. Thus, the next job is begun as the tapes start spinning. The source language is BAL and is designed to work with the tape or disk monitor.

Ordering Procedure: Order File Number 360D-00.3.013.

Distribution will be in card form only.

1130 ERGO: ELEVEN-THIRTY REPORT GENERATING OPERATION. Report Generator for IBM 1130. Single input file, either card or disk. Print and / or punch output. Execution - time I/O overlapped. Specification language reminiscent of 1401 FARGO, but not so limited. Provides alphanumeric manipulation, full arithmetic facility with decimal scaling, user indicators, condition testing with branching, five levels of total, page numbering, etc. Highly diagnostic compiler gives optional program listing. Compile and go operation.

Requires single disk, 8K, 1442, 1132. Written in Assembly language. Runs under Disk Monitor. Occupies five cylinders on disk.

Ordering Procedure: Order File Number - 1130-03.5.001.

Distribution will be in card form only.

IBM-1130 FORTRAN STUDENT SCHEDULING PROGRAMS. This set of 1130 FORTRAN coded programs can schedule 6,000 students based upon 1,000 available course sections of which 270 are different master class courses. Written in three disk-oriented programs, the first creates the disk master schedule file, the second creates the disk student request file, and the third schedules the student requests against the master schedule. A listing is produced for each scheduled student containing information about his shceduled classes and his seat number. The programs are easily modified to meet individual scheduling requirements. This includes 1) increasing the number of students, class courses and sections, and 2) changing the priority inwhich the students are scheduled into their requested courses. An optional program to print the master schedule showing room and final seat allocations is also included. The programs are run under 1130 Monitor (Version 1) supervision. Minimum configuration - CPU 1131-2A/4096 words with disk; 1442-6 card I/O and 1132 printer.

Ordering Procedure: Order File Number 1130-15.4.004.

Distribution will be in card form only.

INTERRUPT SERVICE SUBROUTINE PCCA 1 FOR INCLUSION IN THE IBM 1130 DISK MONITOR SYSTEM FOR HANDLING DATA TRANSMISSION BETWEEN IBM 1130 AND IBM 1070/1050 SYSTEMS. This Interrupt Service Subroutine is to be included in the IBM 1130 Disk Monitor System and handles the communication of the IBM 1130 with Teleprocessing Systems IBM 1070, Model 1or 2, IBM 1050 on one line.

Minimum Machine Configuration: IBM 1130, 4,096 words, one disk drive. IBM 1442 for program input. Communications Channel Adapter CCA (RPQ 835 463, 134 baud, or RPQ 835 464, 6000 baud). External Interrupt (RPQ Y 47 784) is optional.

Core storage required: ISS PCCA 1,784 words plus 40 words in COMMON area plus ca. 10 words for extension of ILS 04.

Ordering Procedure: Order File Number 1130-03.4.004.

Distribution will be in card form only.

1130 NO SPACEBAND. The No Spaceband Program is a modification to the Type Composition Program for the 1130 (1130-DP-04X) Version 1 Mod Level 1, that eliminates spacebands in justified text. This is accomplished by replacing the spaceband with combinations of five fixed width mats. The advantage of this program is increased line caster through put by reducing down time caused by spaceband jams and the elimination of spaceband produced text errors. The program minimizes the use of CN and thin spacequads. The core requirements are 82 positions. No compilation of the Type Composition Program is required. The NOBAND Program is written in assembler language. The 1130 Type Composition program (1130-DP-04X) and the minimum machine configuration for the Type Composition Program are prerequisites for installation of the NOBAND Program.

Ordering Procedure: Order File Number 1130-06.6.008.

Distribution will be in card form only.

IBM Installation Newsletter

April 5, 1968

Issue No. 68-06

CONTENTS

(Green means Green Section)

Distribution: Branch Offices - DP Management, Salesmen, Systems Engineers, FE Managers. Regions, Districts, Education Centers, Field Systems Centers, Federal Systems Centers, FE Area Offices, DPD HQ, WTC.

*Requires Immediate Attention

For IBM Internal Use Only

DPD HQ, White Plains, N.Y.

SELECTED SHORT SUBJECTS

The purpose of Selected Short Subjects is to bring together and highlight concise, factual and timely information which will indicate that action is to be taken by the IBM representative whose accounts are affected.

1. DOS/360 COBOL Version 2 Performance

Apparently, some users are not aware that the performance of DOS/360 COBOL D Version 2 was below that experienced with Version 1. This has been corrected in DOS Release 15, and Field Engineering has PTF's for releases prior to 15. Customers using DOS/360 COBOL should implement Release 15 and take advantage of better performance.

2. S/360 Model 30 24K Incremental Memory

You should be aware of potential programming systems problems on 24K machines. Those which are currently identified are noted below for your guidance.

1. BOS/360, or BPS/360, must be Release 13 or later to be used (See Program Announcement P68-6).
2. The 2030 must be at EC level 126752 or above, and the Micro EC level must be 128063 or above. (P68-6)
3. DOS/360 should be Release 14 or later (Preferably Release 15). Earlier releases will not IPL on a 24K system if they have an 8K or larger supervisor.

3. S/360-20 Submodels 3 and 4 Throughput

There appears to be some confusion regarding internal performance of the new models of the S/360-20. The Product Announcement letter (268-21) indicated that throughput would average 70 percent of that of Submodels 1 and 2. This reduced throughput is not just a reflection of I/O speed but also of CPU instruction execution time.

All models of the S/360-20 have a 3.6 us memory cycle for 1 byte. However, the Submodels' 3 and 4 execution time has been increased. A TNL will be released (and announced in a PRL) shortly that will enable the field to desk time internal execution speeds of the new Submodels.

4. IBM 1800, 1070 and 1080 Assistance Clarification

An article on the above subject was published in issue 68-02 under "Selected Short Subjects", page 2. The author has provided the following clarification which obsoletes and replaces the article in issue 68-02.

The SDD C/DA resident support representative in the 1800 DACS Centers should continue to be the first point of contact for assistance with system design problems, details relative to previously quoted RPQs and sales calls, as outlined in DPD Blue Letter 266-67.

To supplement this service a telephone system has been established with SDD Special Systems Design (Department 590) in San Jose. This system was designed to provide to the field, when directed by the Resident SDD Representative, a rapid response to customer's requests for special equipment and assistance unique to C/DA applications.

Your request, realistically defined and properly channeled, will assure you of receiving the proper assistance.

Telephone: San Jose Tie Line 622, or Area Code 408, 227-7100, Ext. 3714.

OS/360 COBOL F PLANNING INFORMATION

The following is provided for planning purposes only. The actual availability will be as announced through the official IBM procedures --the Program Announcement letter. The availability may possibly be late 1968.

The following improvements are planned for COBOL F:

1. The ability to override DCB parameters at execute time is now available in a limited form for COBOL F (see Page G3 Installation Newsletter 68-02). This will be expanded to include variable length records.

2. A message will be provided for a missing DD card indicating the file, at object time, that did not have a DD card.

3. A condensed listing including only the first instruction for each COBOL verb will be available as a compile time option.

4. Map suppression will provide a compile time option to suppress generating a Procedure Map if an E level diagnostic occurs during compilation.

5. A compile time option will be provided to separate the Procedure and Data Division Maps. You will be able to specify a Procedure Map or a Data Map or both.

OS/360 MVT 2250 SUPPORT PLANNING INFORMATION

The following information is provided for planning purposes only. The actual availability, if any, will be as announced through the official IBM Announcement procedure; the Program Announcement letter.

It is presently planned that the IBM 2250 Display Unit, Model 1 will be supported as the System Operator's Console as an optional feature on Models 50, 65, and 75 of the IBM System/360 Operating System and may be available, for MVT users only, 2nd half of 1968. This support is identical to that provided for MVT with S/360 Model 91 in OS/360 Release 14.

Experience has shown that throughput and operational control are materially improved through the use of this feature.

The 2250 Display Unit can be specified as a primary console at system generation time. It enables the operator to perform the following functions:

1. Observe displayed messages from the system and from the problem program.

2. Reply to messages which require replies.

3. Issue commands to the system.

4. Request optional displays of Unit Status information and Command formats.

If either an IBM 1052 Printer-Keyboard, or a composite console, is specified as an alternate console, a hard copy of commands and messages is provided on the 1052 or the composite console printer.

Use of the 2250 as an operator's console provides the following featues:

1. Faster communications between the operator and the system.

2. A display of more information than is feasible with a typewriter.

Minimum System Requirements:

1. IBM System/360 Computing System, Model 50, 65, or 75 operating under MVT.

2. IBM 2250 Display Unit, Model 1 equipped with the following features:

   A 4096 Byte Buffer
   Light Pen
   Alphameric Keyboard
   Character Generator

Performance

The 2250 Model 1 operator's console support requires 4K bytes storage in the resident nucleus and an additional 4K bytes in fixed main storage for unit status and command format options.

The 2250 Operator's Console support will be documented, until Release 16, with temporary SRL and PLM documentation.

At Release 16, TNL's to the System/360 Operating System SRL and PLM manuals will be available.

S/360-20 DPS MODIFICATION 1 APAR

Modification 1 (P68-26) of S/360-20 DPS contains a character in the label field of the last phase name in the Core-Image Directory that prevents execution of programs whose names are temporarily cataloged. The message OB 22 (phase not found) occurs and the job cannot continue. An APAR has been submitted and a PTF should be forthcoming.

S/360-20 SUBMODELS 3 AND 4 DPS SORT/MERGE ESTIMATES

The Preliminary information in Exhibit 1 is an estimate of the time required to execute the Disk Sort/Merge of the newly announced S/360-20 Submodels 3 and 4.

1130 SYSTEM IMPROVEMENT CONSIDERATIONS

The following article discusses some systems considerations for the 1130.

1. Productive Time Which Cannot Be Improved By Hardware Changes

   Some 1130 system components are available only in one model. The typewriter, the console keyboard, the paper tape reader and the paper tape punch are four such devices (RPQs excluded). The reading/writing speed of the disk is constant. However, because more disk drives may be added, certain other times relative to the disk (seeks, reading of overlays) may be reduced. They are covered in a later section.

2. Productive Time Which Can Be Improved By Hardware Changes

   There are five elements of productive time which may be improved by changing the model or speed of an 1130 component. They are a reduction in PLOTTING time by changing to a faster plotter, or a 2250-4 display tube, and a reduction in CARD READING time, CARD PUNCHING time, and COMPUTING time by changing to faster units.

S/360-20 SUBMODELS 3 and 4 DPS SORT/MERGE ESTIMATES

Time in Minutes

Record Length	File Size	12 K Core 1 Drive	12 K Core 2 Drives	16 K Core 1 Drive	16 K Core 2 Drives
80	5000	7	5	6	5
	10000		9		9
100	5000	7	6	6	5
	10000		12		9
200	5000		9		8
270	4200		15		13

Exhibit 1

a. Plotting

There are two plotting speeds available, tied to carriage sizes. The 1627 Model 1 with an 11 inch carriage is twice as fast as the Model 2 which has a 29 1/2 inch carriage. However, most users have chosen one model or the other on the basis of carriage size rather than speed. A switch to the 2250-4 display tube may be a possibility.

b. Card Reading

There are four different card readers which may be attached to the 1130 system, each with a different card-read time:

Device	Milliseconds per Card
IBM 1442 Model 6	200
IBM 1442 Model 7	150
IBM 2501 Model A1	100
IBM 2501 Model A2	60

If user programs are in IBM-supplied FORTRAN, none of the specified card read time will be overlapped with any other activity.

With a 1442 Model 6 on the 1130, for example, and reading 10 cards, 10 X 200 or 2000 milliseconds will be consumed. This is in addition to whatever manipulation must be performed on the data in those cards. In a FORTRAN program, the system must convert the Hollerith card codes to EBCDIC, break that down according to the specified FORMAT statement, and place the resulting data in the proper core location.

The rated speed of the 1442 Model 6 is 300 cards per minute, but this assumes that the 1130 reads a card every 200 milliseconds. Every card will take 200 milliseconds to read, but the system may not read a card every 200 milliseconds. If intervening operations take 100 milliseconds, it will read one card every 300 milliseconds, yielding a speed of 200 cards per minute.

Rated I/O device speeds are difficult to use when evaluating potential system improvements. One must compare alternatives on the basis of the time per card that the CPU is unable to perform some other operation.

Suppose the user has a 1442-6, and times one of his representative jobs. It reads 2000 cards, and runs for 1200 seconds. The 1130 must have spent 400 seconds reading cards and 800 seconds in computation and other I/O.

If he changed to a 1442-7, the card read time would drop to 300 seconds and the total run time would drop from 20 minutes to 18.3 minutes. The use of a 2501-A1 would cut the time to 16.7 minutes; a 2501-A2 would cut it to 15.3 minutes.

c. Card Punching

There are three different card punches available for use on the 1130 system. All three operate in the same mode, punching one column at a time. The models 6 and 7 both read and punch; the model 5 punches only

Device	Milliseconds per Card Column Punched or Spaced
IBM 1442 Model 6	12.5 plus 12.5 per column
IBM 1442 Model 7	6.5 plus 6.5 per column
IBM 1442 Model 5	6.5 plus 6.5 per column

The overall speed is determined by the last column punched, rather than the number of columns punched. If the user skips the first twenty columns and punches into the 21st, he has punched or spaced 21 columns, and the time for that number will apply.

To continue the previous example, suppose that out of the 2000 cards read, the program punched into the first 20 columns of 500 of them.

For the 1442-6, the breakdown now becomes

Read 2000 cards	400,000	milliseconds
Punch 20 Col., 500 cards	131,250	" "
Compute & other I/O	669,750	" "
TOTAL	1,200,000	milliseconds

With a 1442-7, it would become

Read 2000 cards	300,000	milliseconds
Punch 20 Col., 500 cards	68,250	" "
Compute and other I/O	669,750	" "
TOTAL	1,037,000	milliseconds

or 17.3 minutes

With a 1442 Model 5 or 7 punching, and a 2501 A1 reader, the time would be 15.6 minutes. The 2501-A2 would reduce the time to 14.3 minutes.

d. Printing

Three different line printers may be attached to the 1130 system, each having different print and skip times:

Device	Time in Milliseconds	
	Print 1 Line	Skip 1 Line
IBM 1132	750	16.2
IBM 1403 Model 6	174.6	5
IBM 1403 Model 7	100	5

To illustrate the improvement possible in this area, consider another example. Suppose that a program which is essentially a card listing job, in a half hour, reads 1000 cards, prints 1040 lines, and skips 160 lines. This can be broken down as follows:

	milliseconds
Card Read Time (1442-6)	200,000
Print Time (1132) 1040 @750	780,000
Skip Time 160@ 16.2	2,590
Everything else	817,410
TOTAL	1,800,000 milliseconds (30 min.)

If one replaces the 1132 with a 1403 Model 6, the Print and Skip times drop to

Print (1040 @174.6)	181,600
Skip 160 @5	800 milliseconds

and, added to the card read time and the "everything else" time, results in a total of 1,199,810 milliseconds, about 20 minutes, as opposed to the previous 30 minute figure.

Note that despite this dramatic increase in throughput, the 1403 is printing at only 52 (1040/20) lines per minute, far below its rated speed of 340 lines per minute. The 1132 was also below its rated speed of 80 lpm, since it printed 1040 lines in 30 minutes, or 34.6 lpm.

This shows again that rated speeds of cards per minute, lines per minute, etc., should not be used when investigating alternate approaches to improving throughput. A usable figure is the lenght of time the CPU is busy.

e. Computing

The 1131 Central Processing Unit is available with two basic cycle times- 3.6 microseconds (Models 1 and 2) or 2.2 microseconds (Model 3). In this area it is not quite as easy to calculate the improvement to be expected from the faster CPU. The problem is that one often does not know how much time was spent in computing before, (with a 3.6 microsecond CPU), in which case one cannot accurately determine the effect the 2.2 microsecond CPU will have.

Let's review the previous example: 1442-6 and 1132; 30 minutes run time, read 1000 cards, print 1040 lines, skip 160 lines. The times were:

Card Read	200,000 milliseconds
Print and Skip	782,590 " "
Everything else	817,410 " "
TOTAL	1,800,000 milliseconds (30 Min.)

The only way one could determine the 817,410 value was by subtracting one known value (I/O times) from another known value (total run time).

If one knows that all 817,410 milliseconds were spent in computing, he can calculate that the 2.2 microsecond CPU will do the same amount of work in 61% of that time, or 498,000 milliseconds, a reduction of 5.3 minutes.

If those 817,410 milliseconds had included any disk operations, one could not have made the above estimates, since he would have had no way to determine the split between disk activity and computing. Aside from a good estimate, the only way to evaluate the effect of a new CPU in this case would be to take the program to such an 1130, run it, and time it.

3. Non-Productive Time Which Can Be Reduced By Hardware Changes

By definition, three items fall into this category:

a) DISK seek, to get from one data record to the next.

b) DISK seek, to get from data area to overlay area, and vice versa.

c) DISK read to read overlay.

Note that:

a) will be required whenever data files are used.

b) will be required whenever overlays are used. (SOCALS, LOCALS, AND/OR LINKS);

c) will be required whenever both overlays and data files are used.

The time requirements of all three are difficult to determine, so an exact analysis will not be attempted.

There are two hardware changes which will reduce these times:

1. more core storage, which will eliminate overlays and, therefore, Items b and c.

2. more disk drives, which will allow a redistribution of files and overlays, and reduce items a and b.

a. Additional Core Storage

Aside from programmer convenience, the main advantage in adding more core storage is its probable effect on performance, or run times. If the user can execute programs without any overlays, they can be expected to run at some "top" speed.

b. Additional Disk Drives

Unlike core storage, which will probably be augmented to improve performance, additional disk drives are likely to be considered primarily to increase capability-the capability to copy disks, the additional storage gained, etc. However, in many cases, the move from a single disk 1130 system to a multiple disk 1130 may also be accompanied by a gain in the throughput or performance. Plan the system so that the LOCAL/ SOCAL overlays are on a cartridge other than the one on which the data files reside.

1282 EC'S FOR IMPROVED PERFORMANCE

Since the first customer shipment in 1965, a number of Engineering Changes have been released which will significantly improve the reliability, service-ability and performance of 1282's. Records indicate that many of the 1282's installed are not up to the latest EC

Level. These EC's require a substantial number of hours to install and not every change will significantly affect the performance of all machines. Special effort is needed to coordinate the time with the customer and the IBM Customer Engineer and to install the significant EC's.

Exhibit 1 gives the benefits of the significant changes.

Exhibit 2 lists all the changes and the estimated FE man hours required. Check your customer's machines and establish a plan with FE--to improve performance and customer satisfaction.

NOTE: The "Man Hours Required" shown in Exhibit 2 should be checked with local Field Engineering for accuracy.

1282 Improved Performance

Significant Changes & Improvements Obtained

Thru E.C.

124391	Provides for the safety against erroneous reading of Farrington Font. Especially important where the customer is reading "unchecked" imprinted amounts.
126011	Improves throughput by allowing the C.E. to set the 1282 for only one rescan, but that scan to be at the proper scanning level.
126952	Improves the capability of distinguishing valid scannable data by "blanking" out invalid data, both above and below the scannable characters (available with 7B Font only).
126967	Allows the 1282 to retain successfully read fields of wide density ranges via rescan. Mandatory change for optimum reading of Amount Imprinting all fonts. This has to be ordered as it is a "no cost" feature.
132300	Improves the ability of the 1282 to read degraded 1428E font imprinting.

Customers reading only

Farrington 7B Acct. No.	should be at least up through	E.C. 126952.
Farrington 7B Acct. & Amt.	should be at least up through	E.C. 126967.
1428 Normal Front	should be at least up through	E.C. 126020.
1428E Amt. and/or Acct.	should be up through	E.C. 132300.

Exhibit 1

For IBM Internal Use Only

1282 Improved Performance

Changes and FE Man Hours		Estimated CE Man Hours*	
Eng. Change	Field Bill		Description of Improvement
124759	487631	1.0	Diagnostic
124769	487644	3.5	Farrington Font Substitution
124391	487643	4.5	Farrington Font Substitution
124770	487699	4.5	Non-Process run out
124774	487729	4.5	Maximum punch suppression
125313	487837	4.5	Recognition
125315	487847	5.0	Rescan
126011	488112	1.0	Rescan
126018	488232	2.5	Special Symbol Recognition
125322	488054	5.0	Meter
126020	488274	2.5	1428 Normal Font Recognitio
126019	488389	1.0	Baseline
126959	5870000	3.5	Hi-voltage
126952	5870018	5.5	
	5870019	3.5	Blanking
	5870020	2.5	
126953	5870031	5.0	Clipping Level Control
	5870032	5.0	
126972	5870049	2.5	Recognition
126974	5870062	3.5	Scanner Clutch
130026	5870065	.75	Brush
126002	488094	6.9	Scan Card Lever
130003	5870095	4.1	Power Supply
130029	5870081	13.0	1428 E Font Recognition Part
130032	5870110	8.1	Safety
126967	5870119	8.9	Extended Memory-Amount Imprinting
130039	5870159	3.0	Feed
130028	5870079	7.2	Lamp
130036	5870192	.7	Magnetic Drum
130045	5870223	3.2	Feed
132300	5870200	3.0	1428 E Font Recognition Part

Exhibit 2

* Check with local Field Engineering for Accuracy

TELEPROCESSING - AUTO CALL LIMITATIONS

The use of the following information is subject to IBM Policy as contained in the Sales Manual, page GI 3, under "Planning a Teleprocessing System". You should be aware of this policy.

Automatic Calling Units (ACU) 801 type are provided by the telephone companies to operate in conjunction with various Auto Call Features on the IBM 2701, 2702, 2703, and S/360-20. On each of these machines, the purpose is to automatically accept "dialing" digits from the computer and convert them to "dialing" information for the central office equipment without human assistance in order to establish a call.

Many customers would like to use the above arrangements to place calls over WATS or "tie-lines" through their PABX or Centrex equipment which provides conventional telephone service for the computer site. This has advantages, in that the WATS lines and tie-lines can be accessed by telephones for normal voice use and by the computer for data use. However, if such usage involves dialing a few digits and then waiting for a second dial tone, the ACU801 can not be used. The ACU-801 recognizes a dial tone by actually sensing a grounded lead which occurs simultaneously on the telephone line. Once dialing has started, the ground remains, and the second dial tone is strictly audible. Thus, the ACU will ask for the next digit from the computer immediately after "dialing" the previous digit.

If keeping the flexibility of multiple use is important, there are several possible solutions to this problem:

1. Many Centrex CO installation (not CU) can make toll and tie-line calls without a separate second dial tone. Although this is not a direct answer to the above problem, it is a possible alternative.

2. An (IBM) RPQ has been approved which alters the #1303 Auto Call Adapter (which provides Automatic Calling for the SDAI on the 2701). This RPQ (M36749-Auto Call Unit Digit Delay) delays the presentation of any digit to the ACU if the byte containing the digit also has a 1 in the high order bit position. The delay is adjustable, by the Customer Engineer, from 3 to 6 seconds. A minor problem in using this seems to be that STRAM puts ones in the high order positions of all the digits, but this is easily corrected. Of course, for use on any other Adapter or machine, a new RPQ would have to be engineered.

3. Operator intervention, at the PABX or at Centrex attendant positions could be used to temporarily assign such second dial tone circuits to the computer for direct use. This would probably requires some equipment arrangements with the telephone company. In this way, on a scheduled or demand basis, the stored number to be dialed would omit any "first dial tone digits" normally required to access the second dial tone.

4. The telephone companies have, on occasion, provided a special assembly switching arrangement to permit retermination of the desired line from PABX levels directly to the computer (automatic version of item 3 above).

In considering any of the above solutions, keep in mind that the telephone companies do not recommend data transmission on, or through, PABX's. In addition, any set time delay (if item 2 is used) will not be long enough all the time, and special recovery routines will be needed to avoid constantly redialing into a temporary extra long delay. The only protection, when dialing resumes before dial tone is present, is the ACR timer on the ACU. That time (7-40 seconds) has to completely run, starting after the last digit is presented.

Finally, you should also be alert for the amount of line time actually required on the WATS or tie-lines. A traffic study may show that the computer needs such lines essentially full time. In such a situation, there is no advantage to shared use and direct connection removes the above problems.

For information on obtaining RPQ's, see the Sales Manual, page GI 3, item 5 under "Planning a Teleprocessing System".

SECOM-TYPE III'S IN DEVELOPMENT

Programs that are under development as Type III are no longer included in the TIE index.

Beginning March 20, the former TIE abstract will go into SECOM. As before, submission of the abstract is no guarantee that the program will be completed. A program title, or a brief description of its intent, will be broadcasted and printed along with the normal SECOM abstracts. A more complete description of the program, and the person developing it, may be obtained through a normal SECOM request using a dial-up 1050 or an ITPS request (use ITPS RETAIN/SECOM Inquiry Form 120-1687). ITPS requests may not use the ALL function but must request one text item in each message.

To separate this category from the normal SECOM entry, PROG will be used where the system is normally indicated in the abstract number.

The next position will indicate the Operating System, i.e., N is DOS, S is OS (360N is DOS, 360S is OS).

The final four positions of the abstract ID will be the classification code from the Catalog of Programs in the form XX.Y (XX is major; Y is minor category). For example the request:

SEC PROG N 06.7/001 means SECOM file-Program-DOS-DATA handling-Information Classification and Retrieval-item number 1.

See your SECOM ITPS broadcast and the weekly listings that are mailed to all SE Managers for a list of these programs. Questions or comments may be referred to SECOM Coordinator, DPD HQ, Department 713.

plies. YOUR COOPERATION IN PROPERLY HANDLING THE CONTENT OF THE IBM INSTALLATION NEWSLETTER IS NECESSARY FOR ITS CONTINUED PUBLICATION.

Changes to the quantity of the Newsletter shipped are to be made by the Administration Manager. (NOTE THAT EXTRA COPIES OF THE NEWSLETTER ARE NOT TO BE ORDERED FOR DISTRIBUTION OF INFORMATION TO CUSTOMERS.) He should update and send the DP Administration Managers' Communication Control card (form number N44-780) to DP Communication Control, DPD HQ.

Direct inquiries concerning the contents of this Newsletter to:

W. P. Gaffigan, Editor
IBM Installation Newsletter
Installation Planning-#713
DPD HQ
112 East Post Road
White Plains, New York 10601
Telephone 914 WH9-1900 Extension 4781

INFORMATION ABOUT THE NEWSLETTER

EVERY SALESMAN AND SYSTEMS ENGINEER SHOULD RECEIVE ONE COPY OF THE NEWSLETTER. Each FE Branch Manager should receive five copies for distribution to Customer Engineers.

The IBM Installation Newsletter is distributed only to IBM locations and is not available to customers. The Newsletter is separated into two parts. The first part contains information which is intended only for IBM personnel and the pages are labeled For IBM Internal Use Only. THESE PAGES, OR REPRODUCTIONS OF THEM, ARE NOT TO BE GIVEN TO CUSTOMERS. However, using your own good judgement, you may discuss pertinent information from these pages with a customer. The second part contains information which may be REPRODUCED by the Branch Office at their discretion and given to customers. These pages are not labeled with any restrictive classification. Note that when any abstracts for Type III programs are given to the customer, he must be informed that the disclaimer for Type III programs as contained in the Newsletter, ap-

NEW PUBLICATIONS ANNOUNCED BY PRL'S

The weekly PRL's (Publications Release Letters) are used to insure that all Salesmen and Systems Engineers are aware of new or revised Marketing Publications. Normally, each issue of the Newsletter will contain information from two PRL's, one following the other. The information will be placed in the Newsletter in its original form with no rearrangement of form numbers or titles. It is not intended to replace existing information and distribution sources. You should be certain that you are aware of these sources.

PRL #8 February 23, 1968

REFERENCE SOURCES

FORM NO.	TITLE	DISPOSITION
A26-3565-4	IBM System/360 Model 20 Bibliography	Scrap A26-3565-3 & N20-0361.27
N20-0030-17	Catalog of Programs for IBM System/360 - January 1968 Supplement to C20-1619-4	Scrap N20-0030-16
N20-0031-13	Catalog of Programs for IBM 1130 Computer System & IBM 1800 Data Acquisition - January 1968 Supplement to C20-1630-3	Scrap N20-0031-12
N20-0361.28	System/360 Mod 20 SRL Newsletter Re: A26-3565-4	Scrap N20-0361.27
N20-7080.26	7080 SRL Newsletter Re: A22-6774-2	Scrap N20-7080.25
N20-1800.22	1800 SRL Newsletter Re: A26-5921-2	Scrap N20-1800.21

MARKETING PUBLICATIONS

221-0489-0	IBM Systems Maintenance Management	NEW
221-0490-0	The IBM 2152 Printer-Keyboard	NEW
502-3000-0	The IBM Annual Report - 1967	NEW - Abstract only
520-1916-0	The 1130 Statistical System Promotional Brochure	NEW
520-2053-0	IBM System/360 Model 85...growing on a large scale	NEW
520-2067-0	QUIKTRAN Program Library Bessel Functions	NEW
520-2068-0	QUIKTRAN 2 Numerical Integration	NEW
520-2069-0	QUIKTRAN 2 Functional Integration	NEW
520-2070-0	QUIKTRAN 2 Integration of a First Order Differential Equation	NEW
520-2071-0	QUIKTRAN 2 Numerical Differentiation	NEW
520-2072-0	QUIKTRAN 2 Integration by Gaussian Quadrature	NEW
C26-3715-2	IBM 1130/1800 Basic FORTRAN IV Language	Scrap C26-3715-1
C28-2043-1	IBM System/360 Time Sharing System Addendum	Scrap C28-2043-0
C28-6812-2	IBM System/360 Model 44 Programming System: Guide to System Use	Scrap C28-6812-1 & N28-0560
C28-6814-1	IBM System/360 Model 44 Programming System: Systems Programmer's Guide	Scrap C28-6814-0 & N28-0561
C28-6815-2	IBM S/360 Model 44 Programming System Operator's Guide	Scrap C28-6815-1
C33-6001-2	IBM S/360 Model 20 Disk and Tape Programming Systems I/O Control System for the IBM 1419 and 1259 Magnetic Character Readers	Scrap C33-6001-1
H20-0479-0	Matrix Language System/360 (MATLAN) Application Description	NEW
H20-0489-0	Program for Optical System Design II (POSD) Application Description	NEW
J20-0011-1	Introduction to DATATEXT	NEW - Abstract only
L22-6871-3	IBM S/360 2911 Manual Switching Unit, Models 1, 2, 5 and 2989 Remote Switching Console mod. 8	Scrap L22-6871-2
N20-1051	TNL to General Purpose Simulation System/360 OS (360A-CS-17X) Operator's Manual Re: H20-0311-2	NEW
N20-1076	TNL to IBM S/360 Model 30: 1401 Compatability Mode Operator's Manual Re: C20-1650-1	NEW
N20-1849	TNL for System/360 Model 20, Wholesale IMPAC I Program Library - Application Description Re: H20-0461-0	NEW
N21-0088	IBM 1287 Optical Reader Re: A21-9064-1 with N21-0086	NEW
N23-0615	ENL to IBM S/360 PL/I Coding Intro. to PL/I Exam (P.I.) Re: R29-0114-0	NEW
N24-5228	TNL to IBM S/360 BPS I/O 1412/1419 Specs. & Operating Guide Re: C24-3398-4	NEW
N27-3023	Changes to 2701 Component Description Re: A22-6864-2	NEW
N28-0570	IBM S/360 Model 44 Programming System Concepts and Facilities Re: C28-6810-1 with N28-0562	NEW
N33-8527	S/360 Mod 20 TPS Sort/Merge Program Re: C26-3804-1 with N33-8519	NEW
N33-9016	TNL to S/360 Mod 20 DPS Op. Proc. Re: C33-6004-0	NEW
N33-9017	TNL to Mod 20 DPS IOCS Re: C24-9007-2	NEW
N33-9019	TNL Mod 20 DPS Performance Re: C33-6003-0	NEW
N33-9020	TNL to Mod 20 DPS Disk Sort/Merge Re: C26-3806-1 with N33-9013	NEW
R20-0066-0	Property and Liability Information System Education Material - Volume 5 Record Formats - Automobile	NEW
R20-1062-0	IBM System/360 OS/DOS Communications Coding QTAM Coding Flowchart Self-Study	NEW
R20-4091-0	1287 Optical Reader Education Guide	NEW
R20-9196-0	IBM S/360 Operating Systems Facilities General Purpose System Simulator Course Description	NEW
R20-9197-0	IBM S/360 General Purpose Systems Simulator - Advanced Course Description	NEW
R20-9214-0	IBM 1287 Optical Reader Course Description	NEW
R29-0070-2	IBM BCSP P.I. Advisor Guide	Use R29-0070-1
Y20-0126-0	Project Control System/360 (360A-CP-06X) System Manual	NEW
Y28-6814-1	IBM S/360 Model 44 Programming System: Utilities and Stand-Alone Programs -- PLM	Scrap Y28-6814-0
Y28-6822-0	IBM S/360 Model 44 Programming System: Supervisor and Job Control PLM Re: Y28-6812-0 with Y28-6818	NEW
Y28-6823-0	IBM System/360 Model 44 Programming System: Linkage Editor PLM Re: Y28-6813-0	NEW
Y28-6824-0	IBM S/360 Model 44 Programming System: Assembler PLM Re: Y28-6811 with Y28-6817	NEW

PRL #8 (Continued)

Y33-8505-0	TNL to Model 20 TPS RPG PLM Re: Y33-9001-0	NEW
Y33-9009-1	IBM S/360 Mod 20 Disk and Tape Programming Systems I/O Control System for the IBM 1419 Magnetic Character Reader PLM	Scrap Y33-9009-0
Y33-9021-0	TNL to Mod 20 DPS IOCS PLM Re: Y33-9007-0	NEW
Y33-9022-0	TNL to Mod 20 DPS/TPS RPG PLM Re: Y33-9015-0	NEW
Y33-9023-0	TNL to Mod. 20 DPS Disk Sort/Merge PLM Re: Y33-9016-0	NEW
Z28-6636-3	IBM S/360 CLEAR Programmer's Guide	Scrap Z28-6636-2
Z77-8011-0	Intermediate Work Area for User Programs Written in DOS FORTRAN	NEW
Z77-8016-0	Compute Withholding Taxes for 50 States, District of Columbia and the Federal Government in RPG 8K Mod. 20 Card System	NEW

PRL #10 March 8, 1968

REFERENCE SOURCES

A24-3089-6	Teleprocessing and Data Collection Bibliography Supplement	Scrap A24-3089-5
N20-0360.45	S/360 SRL Newsletter Re: A22-6822-10	Scrap N20-0360.44

MARKETING PUBLICATIONS

220-1921-1	System Rental Plan	NEW
221-0345-3	The IBM 2050 Processing Unit Proposal Insert	Scrap 221-0345-2
221-0482-0	The IBM Linear Programming-Mathematical Optimization Subroutine System (LP-MOSS)	NEW
320-1906-1	In Brief, January 1968	Scrap 320-1906-0
520-2017-0	A Programmed Airlines Reservation System for the Jet Age	NEW
570-0337-1	Micro Records Systems Federal Supply Price Schedule	Scrap 570-0337-0
570-0208-7	Term Contract for Tabulating Cards	Scrap 520-0208-6
A26-5889-2	IBM System/360 Model 20 Introduction and System Summary	Scrap A26-5889-1
A27-2715-1	IBM System/360 Special Feature Description 709/7040/7044/7090/7094II Compatibility Feature for System/360 Model 65	Scrap A27-2715-0
C20-1618-3	Number Systems Student Text	Use C20-1618-2
C24-5061-2	IBM System/360 Basic Programming Support Basic Tape System System Generation and Maintenance	Scrap C24-5061-1 & N24-5236
C26-3720-0	IBM 1800 Multiprogramming Executive Operating System Programmer's Guide	NEW
C26-3750-2	IBM 1130 Disk Monitor System Reference Manual	Scrap C26-3750-1 & N26-0540-0
C27-6918-1	IBM S/360 Operating System: Maintenance	Use C27-6918-0 & N28-2160
C28-2023-0	IBM System/360 Time Sharing System Master Index	NEW
C28-6644-1	IBM System/360 Operating System: Master Index	Scrap C28-6644-0
H20-0491-0	AUTOSPOT II (1620-CN-05X) Version 2 Part Programming Manual	NEW
K20-0252-0	Preparation of Client Financial Statements Using an IBM 1130 Computing System at the CPA Firm of J. K. Lasser & Company	NEW - Scrap Y20-0125
N20-1079-0	TNL to A Programmer's Introduction to the IBM System/360 Architecture, Instructions and Assembler Language Re: C20-1646-4 with N20-1069	NEW
N24-5322	TNL to IBM S/360 BPS Basic Tape System Operating Guide Re: C24-3391-3	NEW
N24-5327	TNL to S/360 BPS Basic Tape System Programmer's Guide Re: C24-3354-6 with N24-5339	NEW
N24-5342	TNL to S/360 BPS Assembler with I/O Macros Specifications Re: C24-3355-5	NEW
N24-5353	TNL to IBM S/360 BPS/BTS Operating Guide Re: C24-3391-3 with N24-5322	NEW
N33-8530	TNL to IBM S/360 Basic Assembler (Card), Operating Procedure Re: C26-3802-2	NEW
N33-9024	TNL to S/360 Mod. 20 DPS Disk Sort/Merge Re: C26-3806-2, -1	NEW
N33-9026	TNL to Mod 20 DPS/TPS Assembler Language Re: C24-9002-3	NEW
R20-4014-0	IBM S/360 OS QTAM Coding Learner's Guide	NEW
X20-1703-5	IBM System/360 Reference Card	Scrap X20-1703-4
X24-6514-4	Card Proving Machine - Card Punches - Verifies - Tape Punches Physical Planning Template	Scrap X24-6514-3
Y20-0160-0	AUTOSPOT II (1620-CN-05X) Version 2 System Manual	NEW
Y27-7110-1	IBM S/360 Operating System Graphic Problem Oriented Routines	Use Y27-7110-0, Y27-7141 & Y27-7150
Y33-8001-0	IBM S/360 OS PLM TNL Re: Y33-8000-0	NEW

IBM INSTALLATION NEWSLETTER
TABLE OF CONTENTS

OS/360 MVT CPU TIMING ALTERATION

The following has not been submitted to any formal IBM test. Future releases of OS/360 may obviate the need for this alteration. Potential users should evaluate its usefulness in their own environment prior to implementation.

1. The module that is affected by this alteration is the MVT Dispatcher, which is assembled as part of the Nucleus. The name of the Nucleus macro in SYS1. GENLIB is IEAQNU.

2. The user should follow this procedure to assure that he is dealing with a current level of the system and not introducing "down level code" into his system.

3. INSTRUCTIONS:

   a. Punch out the IEAQNU macro from SYS1. GENLIB. This is approximately 2500 cards.

   b. CONSULT THE SYSGEN LISTING OF YOUR SYSTEM! There, you will find where the nucleus is assembled. The IEQANU macro will be there with several parameters dealing with the timing option selected during SYSGEN, probably JSTIME, the CPU model of the machine, which SER option was selected etc. The user must punch another card EXACTLY like the one produced by his SYSGEN with the exception of the first (or timing option) field. This must be punched with TIME6B. Then the cards shown in Exhibit 1 are inserted in the dispatcher part of the IEAQNU macro at approximately statement number 0942000 "TIMER DEQUEUE FOR TASK INTERVALS" (#1), and at statement number 10140000 "TIMER QUEUEING FOR TASK INTERVALS" (#2). Then assemble this macro using the IEAQNU TIME6B... statement that the user has prepared. Linkedit the resultant object module into the Operating System SYS1. NUCLEUS data set, again using the Linkedit from the SYSGEN as a pattern. BE SURE TO INSERT IEAANIPO as the first load module and IEAQFX00 as the second.

4. The system is now ready to be re-IPL'ed with the new timing algorithm.

5. The user should be aware that this alteration will produce shorter times than his previous ones. However, these times will be more consistent since they represent only the CPU active time and none of the WAIT time. The ability to "time-out" the job that goes into an endless WAIT has been lost.

OS/360 MVT CPU TIMING ALTERATION

```
* TIMER DEQUEUE FOR TASK INTERVALS.                              09420000
*       *****BEGINNING OF CPU TIMING PATCH*****
          L XDSTQE,TCBJSTCB(0,XDSOLD) .GET JOB STEP TASK
          LTR   XDSTQE,XDSTQE    .IS THIS THE WAIT TASK
          BZ    PATCH1
          L XDSTQE,TCBOTC(0,XDSTQE) .GET INIT TCB
          LTR   XDSTQE,XDSTQE  .IS THIS THE MASTER SCHEDULER
          BZ    PATCH1
          L XDSTQE,TCBTME(0,XDSTQE) .GET INIT TQE
          LTR XDSTQE,XDSTQE .DOES INIT HAVE A TQE
          BZ PATCH1
          TM TQEFLGS(XDSTQE),X'8B' .IS THIS FOR TASK TIME
          BC 5,PATCH1 .GO IF NO
          L R15,ATDQ
          BALR XDSLNK,R15 .GO TO TIMER DEQUEUE ROUTINE
          USING *,XDSLNK .REESTABLISH
          L 14,ADSBASE .ADDRESSABILITY
          DROP XDSLNK .
          USING DSBASE,14 .
PATCH1    EQU *
*       *****END OF CPU TIME PATCH*****
```

#1

OS/360 MVT CPU TIMING ALTERATION (Continued)

```
         L     XDSTQE,TCBTME(0,XDSOLD) .GET TQE OF 'OLD' TASK.          09440000
         LTR   XDSTQE,XDSTQE           .DOES TQE EXIST                  09460000
         BZ    DSTEST                  .BR IF NO TQE.                   09480000
         TM    TQEFLGS(XDSTQE),X'8B'   .IS TQE ON QUEUE AND TASK TYPE.  09500000
         BC    5,DSTEST                .BR IF NOT.                      09520000
         L     R15,ATDQ                GET TIMER DEQUEUE ADDR.          09540000
         BALR  XDSLNK,R15              TO TIMER DEQUEUE RTN             09560000
         USING *,XDSLNK                                                 09580000
         L     14,ADSBASE              .REESTABLISH ADDRESSABILITY      09600000
         DROP  XDSLNK                                                   09620000
         USING DSBASE,14                                                09640000
* TEST FOR READY TASK.                                                  09660000
.TESTR   ANOP                                                           09680000
&TAG     LTR   XDSNEW,XDSNEW       .IS NEW READY TASK FOUND.            09700000
         BZ    DSEARCH                 .BR IF NOT.                      09720000
&TAG     SETC  'DSREADY'                                                09740000
         AIF   (&MAINLNE EQ 1).SETOLD                                   09760000
         AIF   (&JSTIME NE 1).SETOLD                                    09780000
* JOB STEP TIMING CODE                                                  09800000
         LR    XSEARCH,XDSNEW          .SET START TCB ADDR FOR CONTROL  09820000
*                                       GOING TO HIGHER PRIORITY TASK.  09840000
         LR    XLIMIT,XDSOLD           .SET END TCB                     09860000
         L     R15,ATDQ                GET TIMER DEQUEUE ADDR.          09880000
         BAL   XSBR,DJSEARCH           .TO JST SUBROUTINE.              09900000
         L     XSEARCH,A1STCB          .SET START TCB ADDR FOR          09920000
*                                       SEARCHING TCB LIST FROM TOP.    09940000
&TAG     LR    XLIMIT,XDSNEW                                            09960000
&TAG     SETC  ''                                                       09980000
         L     R15,ATNQ                GET TIMER ENQUEUE ADDR.          10000000
         BAL   XSBR,DJSEARCH           .TO JST SUBROUTINE.              10020000
* SET 'OLD'   'NEW'.                                                    10040000
.SETOLD  ANOP                                                           10060000
&TAG     LR    XDSOLD,XDSNEW                                            10080000
         STM   XDSNEW,XDSOLD,IEATCBP                                    10100000
         AIF   (&MAINLNE EQ 1).DS02A                                    10120000
* TIMER QUEUING FOR TASK INTERVALS.                                     10140000
*      *****BEGINNING OF CPU TIMING PATCH*****
         L XDSTQE,TCBJSTCB(0,XDSNEW) .GET JOB STEP TASK
         L XDSTQE,TCBOTC(0,XDSTQE) .GET INIT TCB
         LTR   XDSTQE,XDSTQE  .IS THIS THE MASTER SCHEDULER
         BZ    PATCH2
         L XDSTQE,TCBTME(0,XDSTQE) .GET INIT TQE
         LTR XDSTQE,XDSTQE .DOES INIT HAVE A TQE
          BZ PATCH2
         L R15,ATNQ
         BALR XDSLNK,R15 .GO TO TIMER ENQUEUE ROUTINE
         USING *,XDSLNK .REESTABLISH
         L 14,ADSBASE .ADDRESSABILITY
         DROP XDSLNK .
         USING DSBASE,14 .
PATCH2   EQU *
*      *****END OF CPU TIME PATCH*****
```

#2

Exhibit 1

OS/360 DATA RETRIEVAL FROM DASD

The following article is from an IBM Field Systems Center and was authored by an IBM Field Engineering Specialist.

It discusses data retrieval from direct access storage devices and the importance of data set specifications in obtaining efficiency.

A slow data retrieval rate from a direct-access storage device often occurs when proper attention is not given to data set specifications. This article addresses in particular, the record format specification as selected by the RECFM parameter in the DCB macro. See the SRL publication, "Operating System/360 Supervisor and Data Management Macro-Instructions", Form C28-6647.

To obtain efficient data retrieval rates, use a "fixed" or "fixed-blocked" format wherever practical. When ever "fixed" or "fixed-blocked" is practical, always use "fixed-standard" or "fixed-blocked standard" unless unacceptable for the reasons given below. A "standard" format data set contains no embedded short blocks or embedded "empty" tracks. Therefore, "head-switching" (switching to the next track) requirements can be predicted, based on the block size of the data set and the track capacity of the particular I/O device. "Standard" access module coding provides "automatic" head-switching. In contrast, the access module coding for "non-standard" data sets cannot predict head-switching requirements; hence, the next record to be read is assumed to be on the same track as the record previously read. If the next record is not on the same track as the previously read record, an I/O interrupt occurs, and a "unit-check" status is presented. This results in a "sense" command being issued to the interrupting device, and the device-dependent error routines to be entered. The error routines analyze the conditions, perform a head-switch, and restart the operation, attempting to read a record from the next track. Obviously, use of a "non-standard" specification may needlessly degrade performance. Degradation becomes severe when the device-dependent error routines are non-resident, i.e., they must be fetched into main storage each time they are required.

A striking example of an unnecessarily poor data retrieval rate involved the creation of a fixed-blocked data set on a 2321 Data Cell Drive, specifying full-track blocking. The time required to retrieve this data set was approximately three times that required to create it. A satisfactory retrieval rate was obtained by specifying the record format as "fixed-standard" when retrieving the data set. Note that any "fixed" or "fixed-blocked" data set may be retrieved as a "standard" data set even though it was not created as a "standard" data set, as long as the data set contains no embedded short blocks or empty tracks.

Other procedures that may improve performance are:

1. Creating data sets on cylinder boundaries, particularly partitioned data sets.
2. Obtaining contiguous tracks for a data set.
3. Avoiding secondary extents (secondary allocation).
4. Assuring that no high-usage data sets have extents encompassing tracks with alternate assignments.
5. Using full-track blocking where practical.
6. Using the chained-scheduling option.

OS/360 AIDS IN USING COBOL F SORT FEATURE

The following contribution has been tested by the author. It has not been submitted to any formal IBM test. Potential users should evaluate its usefulness in their own environment prior to implementation.

1. How to CALL A Subprogram From an Input or Output Procedure

   A CALL to a subprogram from an input or output procedure is not allowed in the COBOL F language. However, there is a simple technique that will enable CALLs to be made successfully.

   A CALLed subprogram would normally have initialization coding of the form shown in Exhibit 1.

OS/360 AIDS IN USING COBOL F SORT FEATURE

```
            STM     14, 12, 12(13)      Save COBOL regs
            ST      13,SUBSAVE+4        Save pointer to COBOL save area
            LR      12, 13              Transfer the pointer to a work reg
            LA      13,SUBSAVE          Load Addr of subprogram save area
           *ST      13, 8(12)           Store subprogram save area pointer in
            ...                         COBOL's trace forward cell
SUBSAVE     DS      18F
```

Exhibit 1

When an input or output procedure receives control, the COBOL save area trace-forward cell points to the save area of the COBOL library subroutine, IHDFSORT, which interfaces with the SORT program. This pointer must remain intact for proper return linkage. However, the STORE instruction shown above with the asterisk destroys this pointer.

The solution is to omit this instruction if the subprogram is written in Assembly language, or to NO-OP the generated instructionif the subprogram is written in a higher level language. In the case of COBOL F, it appears as the first instruction in the initialization section called INIT2.

2. How to Locate, in a Core Dump, the Last Record Released to SORT By an Input Procedure

   a. From the Data Division map, determine the BLL number of the SORT-file being processed at the time of program termination. Assume it is BLL n.

   b. From the Task Global Table map, determine the location of the BLL cells in the COBOL object program.

   c. The nth BLL in the core dump will point to the last record released to SORT.

   Note that this BLL is initialized when control is first transferred to the input procedure. Thus, if the program terminates before control ever goes to the input procedure, the BLL will not be initialized.

OS/DOS/360 QTAM SELF-STUDY TEXT

If you are planning to install QTAM, or are considering QTAM for the installation of a T/P System, you and your customer should review a new self-study text. The title is "System/360 OS/DOS Communications Coding QTAM Coding Flowchart". The publication number is R20-1062-0 and is available from the IBM Distribution Center, Mechanicsburg.

The QTAM FLOWCHART is a means of learning how to construct a QTAM message control program. Each macro is discussed, in the sequence which might be used to construct an actual program. In fact, as you go through the macros, you are constructing an actual program.

Decisions as to how to select certain macros, or parameters to use in the macros, are put in flowchart format so that you can skip information not pertinent to your particular message control application. Choices of parameters, or macros, are discussed in an introductory manner.

Examples of each new concept are embedded in the flow. These are for the most part DOS examples, actually assembled to show the program generated code so that the student can go into more detail on his own if this is his option. The entire flow was in fact a program assembly. Thus, a decision had to be made whether to make the examples DOS, or OS. Wherever OS and DOS QTAM differ in macro selection or parameter selection, the flow is split down the middle with OS on the left, DOS on the right. Where they are the same, the flow goes down the middle of the page. OS examples are usually given as comments cards (as far as the assembly is concerned).

DOS/360 QTAM-INTERVAL TIMER ASSIGNMENT

Current DOS/360 SRL's specify that the Interval Timer must be assigned to F1 if a polling interval is specified in the DTFQT for any line group. Users should be aware that it is also required that the Interval Timer be assigned to F1 if either, CPINTVL = integer is specified in the TERMTBL macro, or the INTREL function is to be used in operator control.

DOS/360 IGNORE OPTION CORRECTION

An article "DOS/360 Ignore Option in Assign Statement" appeared in Installation Newsletter issue 68-04, page 5.

The statement was made that "if you are rerunning to recreate a report, you can ignore any tapes that are not required". This is not true if you are using LIOCS.

The S/360 DOS Users Guide; Control Statement Techniques (C20-1685-0) states, on page 13; "IGN cannot be used if information from the device is required for proper operation. Therefore, use of IGN results in JOB cancellation at OPEN time if tape or DASD files are processed by LIOCS".

Also, the statement regarding the DOS/360 MERGE under "Sequence Checking" is true only in the case of unlabeled files because LIOCS OPEN is used.

DOS/TOS/360 COBOL AL ROUTINE FOR OVERLAYS

The following is a suggested technique (but not the only one possible) for using an Assembler Language routine to effect overlays with DOS/TOS/360 COBOL. Potential users should evaluate its usefulness in their own environment prior to implementation.

This article contains:

1. An example of a printout of an Assembler Language effecting overlays specified by a COBOL Disk and Tape Operating Systems program.

2. Explanations of the functions performed by the Assembler overlay subroutine instructions. The explanations are keyed to the instructions in the listing.

3. Information needed to prepare and use subprograms written in Assembler language with a main program written in COBOL.

Exhibit 1 contains an overlay subroutine which is an example and is governed by the following restrictions:

1. The example is a suggested technique, and not the only technique.

2. It can be used for Assembler overlays if the user has a desired entry point in his end card, and the first statement at that entry point is 'STM 14, 12, 12(13)'(90ECDOOC).

3. The subroutine cannot be used for entry points other than at the first instruction of the procedure division. A suggested technique for diverse entry points is a table lookup employing V-type constants.

DOS/TOS/360 COBOL AL ROUTINE FOR OVERLAYS

```
STMNT     SOURCE STATEMENT

0001      OVRLAY  START 0
0002              ENTRY OVRLAY
0003      * AT ENTRY TIME
0004      *     R1 = POINTER TO ADCON LIST OF USING ARGUMENTS
0005      *          FIRST ARGUMENT
0006      *          IS PHASE OR SUBROUTINE NAME, MUST BE 8 BYTES
0007      *     R13= ADDRESS OF SAVE AREA
0008      *     R14= RETURN POINT OF CALLING PROGRAM
0009      *     R15= ENTRY POINT OF OVERLAY PROGRAM
0010      * AT EXIT
```

Exhibit 1

DOS/TOS/360 COBOL AL ROUTINE FOR OVERLAYS (Continued)

```
STMNT          SOURCE STATEMENT

0011           *    R1=  POINTER TO SECOND ARGUMENT OF ADCON LIST
0012           *         OF USING ARGUMENTS
0013           *    R14= RETURN POINT OF CALLING PROGRAM--NOT THIS PROG
0014           *    R15= ENTRY POINT OF PHASE OR SUBPROGRAM
0015           *
0016                     USING *,15

STMNT          SOURCE STATEMENT

0017                STM      0,1,20(13)        SAVE REG 0 AND 1
0018                L        1,0(1)            R1 = ADDRESS OF PHASE NAME
0019                CLC      0(8,1),CORSUB     IS IT IN CORE
0020                BE       SUBIN             YES
0021                MVC      CORSUB(8),0(1)    NO, CORSUB = PHASE NAME
0022                SR       0,0               R0 = 0
0023 *                                         LOAD REQUIRES R0 = 0 IF LOAD
0024 *                                         ADDRESS
0025 *                                         ISN'T SPECIFIED, R1=ADDRESS OF
0026 *                                         PHASE NAME. R1=PHASE ENTRY
0027 *                                         UPON RETURN.
0028                SVC      4                 LOAD PHASE
0029                SH       1,=H'2'
0030 ENTRY         LA       1,2(1)            STEP SEARCH POINTER
0031                CLC      0(4,1),STMINS     IS THIS THE ENTRY POINT
0032                BNE      ENTRY             NO, LOOP BACK
0033                ST       1,ASUB            YES, SAVE IT
0034 SUBIN         LM       0,1,20(13)        RESTORE REG 0 AND 1
0035                LA       1,4(1)            STEP PAST PHASE NAME ADCON
0036                L        15,ASUB           LOAD ENTRY POINT ADDRESS
0037                BR       15
0038 ASUB          DS       IF
0039 CORSUB        DC       8X'FF'
0040 STMINS        DC       X'90ECD00C'
0041                END
```

Exhibit 1

S/360-20 DPS-BAL AND USE OF DTFEN OVLAY

The following is a field contribution which has not been submitted to any formal IBM test. Potential users should evaluate its usefulness prior to implementation. The article concerns a correction to the S/360-20 DTFEN Overlay routine and some tips for using S/360-20 BAL OPEN routines.

When certain file characteristics are met, the DTFEN with OVLAY function may be used to conserve core. The use of this technique simply allows the Assembler to generate OPEN and CLOSE coding at the point where these macros are issued --instead of in the IOCS DTF coding.

The following hints are offered to those using this function:

1. An error in the SRL 360/20 DPS I/O Control System, C24-9007-2 will not allow the programmer's second (main logic) phase to be called in during program execution. In figure 20, page 79, each of the two fetches should have blank operands (the blank operand indicates that the fetch coding relates to sequential retrival of subphases). See Exhibit 1.

2. If card files are used, the I/O AREA must be available to IOCS at the time of OPEN. At OPEN time the first card is read into the I/O AREA (unless no overlap is specified). The I/O AREA must, of course, also be available to any other phase that uses the card I/O AREA. Therefore this I/O AREA should normally be specified at the beginning of the program before the OPEN routine. In this way the OPEN and the remainder of the first sub-phase will be overlayed, but the I/O AREA will not be overlayed. See Exhibit 1.

   A second option allows the programmer to issue the CARD file OPEN in the main logic phase. Though this method would use main logic core for the OPEN, the card OPEN routine generates substantially less core than the disk or tape OPEN routines.

3. Base register usage above 4K (a second base register) in the first (OPEN) phase has resulted in a programming error stop due to improper handling of the base register by IOCS. The exact extent of the problem has not been determined as yet. However, the first phase has been successfully executed by keeping the first phase to one base register and using register 9 (with limitations as indicated on p. 80, C24-9007-2, TNL N33-9014 as the base register of the first phase.

S/360-20 DPS - BAL AND USE OF DTFEN OVLAY

FIGURE 1

```
                        START
                        DTF . . .
                        .
                        DTFEN       OVLAY
BEGIN                   BASR        9, Ø
                        USING       BRANCH, 9
BRANCH                  B           OPENIT
                        ----        DS FOR CARD FILES
                        ----
OPENIT                  OPEN        CARDLE . . .
                        ----
                        ----
                        FETCH       BEGIN
-------------------------------------------------------------
                        ORG         OPENIT
ROUTIN                  BASR        1Ø, Ø
                        USING       *, 10
                        USING       BRANCH, 9
                        ----
                        ----
                        FETCH
                        XFR         ROUTIN
-------------------------------------------------------------
                        ORG         XXXX (ROUTIN IS
                                    SUGGESTED AS OPERAND)
FINIS                   BASR
                        USING . . .
                        CLOSE . . .
                        ----
                        EOJ
                        END         FINIS
```

Exhibit 1

S/360-20 DPS RPG MULTIPLE CHAINING FILE SEARCH

The following is a field contribution which has not been submitted to any formal IBM test. Potential users should evaluate its usefulness prior to implementation. Addresses, tables, files, indicators, etc. produced in the program generation have been deleted to conserve space.

This technique makes it possible for an RPG user to generate his own chaining field and use it to search an indexed sequential file for a match. The RPG user generates the key and places it in a field called "BOX", (Exhibit 2) then an exit to an Assembler routine (GET, Exhibit 1) takes place where "BOX" is placed in the input work area, overlaying the C1 field data of the last card read. The program then branches to the total processing routine (and total output) after which it extracts the "C1" field from the card input area, uses it to search for the matching disk record, and returns to detail calculations. Note that the core locations for card columns of the input card are located beginning at the input work area plus1.

S/360-20 TPS RPG IMPROPER FIELD NAMING

Field names used as result fields on Calculations Specifications cannot begin with TAB, except LOKUP operations.

Page 164 of C24-9001-3 states that "None of the Field names used on the Input Specifications form should begin with the character combination TAB". The manual should also state that such field names cannot be used for the result field on the Calculations Specifications.

When a TABXXX is used as a result field, there is no diagnostic and the field named TABXXX is ignored and not compiled by RPG. When the program is executed, a specification error occurs.

S/360-20 DPS RPG MULTIPLE CHAINING FILE SEARCH 01/01/68 PAGE 001

```
LOCATN  OBJECT CODE   ADD1 ADD2 STMT     SOURCE STATEMENT                    DPS ASSEMB 01/00

0000                           0001 GET      START 0
0000 0DF0                      0002 BEGN     BASR  15,0                                  003
0002                           0003          USING *,15
                               0004          EXTRN BOX
0002 4090 F016       0018      0005          STH   9,R9                             003
0006 4890 F014       0016      0006          LH    9,ADB                            003
                               0007 ***  MOVE FROM BOX TO C1 FIELD (COLS 1-8) WHICH BEGINS AT
                               0008 ***       INPUT WORK AREA PLUS 1.
                               0009 ***  3D3C HEX TRANSLATES TO 3388 PLUS REG 3.
000A D207 3D3C 9000 0D3C 0000  0010          MVC   3388(8,3),0(9)                   003
0010 4890 F016       0018      0011          LH    9,R9                             003
0014 07FE                      0012          BCR   15,14                            003
0016 0000                      0013 ADB      DC    Y(BOX)                           003
0018                           0014 R9       DS    H
0000                           0015          END   GET
```

Exhibit 1

```
// LOG    MULTIPLE CHAINING TEST
// JOB RPG,GO
// DATE 67013
// CONFG 001
// ASSGN SYSIPT,X'100',R4
// ASSGN SYSLST,X'200',R7
// ASSGN SYS000,X'802',D3
// VOL SYS000,WORK1
// DLAB  'WORK FILE FOR RPG                          1202021',        X
               0001,68013,68013
// XTENT 1,001,0070000,0071009,'202021',SYS000
// ASSGN SYS006,X'802',D3
// VOL SYS006,ISFILE
// DLAB  'INVENTORY FILE                             1202021',        X
               0001,67001,67001
// XTENT 1,001,0082000,0082009,'202021',SYS006
// XTENT 4,002,0080001,0080001,'202021',SYS006
// EXEC
```

Exhibit 2

S/360-20 DPS RPG MULTIPLE CHAINING FILE SEARCH

```
SYSTEM /360 MODEL 20 DPS        RPG (VERS. 0,MODIF. 00)          DATE 01/13/67

S   1       H   M 016        120

S           F****   MULTIPLE CHAINING TEST                      1-16-68
S   2       FCARDS   IP  F  80   80            EMFCM1
S   3       FISFILE  IC  F 540   60R08KI   0002 DISK11FSYS005S              102
S   4       FPRINT   O   F 120 120             PRINTER

S   5       EAAC1CARDS   ISFILE

S   6       ICARDS   AA  01  80 CC
            *********    THE 'GET' SUBROUTINE WILL OVERLAY THIS C1 FIELD FROM
S           **********    THE CONTENTS OF 'BOX'.
S   7       I                                              1   8 KEY      C1
S   8       IISFILE  BB  05

S   9       I                                              1  60 DATA

S  10       C  N88                  MOVE KEY        BAG     8
S  11       C  N88                  MOVE KEY        NUM     80
S  12       C                       SETON                      88
S  13       C   01                  MOVE BOX        BAG
S  14       C   01                  SETOF                      55
S  15       C   01        COUNT     ADD  1          COUNT   50
S  16       C   01 05                SETON                      55
   17   04  C             NUM       ADD  1          NUM
   18   05  C                       MLLZOO          NUM
   19   06  C                       MOVE NUM        BOX     8
   20   07  C                       EXIT GET
S  21       C   01                  GOTO LOOP
S  22       CLO           LOOP      TAG
S  23       C                       RLABL           BOX

S  24       OPRINT   H  2     1P
S  25       O                                       17 'KEY FIELD'
S  26       O                                       25 'DATA'
S  27       O                                       88 'COUNT
S  28       O        T  1     LO
S  29       O                          BAG          17
S  30       O                          COUNT Z      88
S  31       O                   55     DATA   B     81
```

(ASSIGNED AREAS, etc. Then print out not shown in this article.)

Exhibit 2 (Continued)

NEW TYPE III PROGRAMS *

The following are the abstracts of programs which have been recently made available from the Type III library.

The program, along with its complete abstract, will be incorporated in subsequent issues of the Catalog of Programs.

Programs may be obtained by submitting a properly completed "General Program Request Card" (Form Number 120-1145-1) to the Program Information Department, 40 Saw Mill River Road, Hawthorne, New York, 10532.

These programs and their related documentation are distributed by IBM in the author's original form and have not been subjected to any formal testing.

Any discussion of Type III Programs must emphasize the following points:

1. This is a Type III program.
2. **Type III programs have not been subjected to any formal product test.**
3. **Recipients of Type III programs are expected to make the final evaluation as to the usefulness of the programs in their own environment.**
4. **There is no committed maintenance for Type III programs. However, any changes the author chooses to make will be announced in subsequent issues of the Catalog of Programs.**

* **NOTE: THE CUSTOMER MUST BE INFORMED THAT THE ABOVE APPLIES TO ANY OF THE FOLLOWING TYPE III ABSTRACTS FURNISHED TO HIM.**

SYSTEM/360 MULTIPLE LIST PROGRAM. This program permits multiple 80/80 listing of a card deck which is read into the computer only once. The first listing is produced concurrently as the card deck is being read in. Subsequent listings are obtained simply by pressing the interrupt key after the system enters the wait state. It no longer becomes necessary to reload the card deck each time, thus, card-handling, card-read-time, and reader-ready-time are eliminated for extra lists.

The program consists of four cards and is written in machine language for any model S/360 with a 2540 Card Reader and a 1403 Printer on line.

The card images are stacked into core, so 32K will accommodate a card deck of about 400 cards, 64K-800 cards, etc.

Ordering Procedure: Order File Number 360D-06.8.001.

Distribution will be in card form only.

SYSTEM/360 REPRODUCE AND SEQUENCE-NUMBER UTILITY PROGRAMS. This is a package of three 2-card programs. One will reproduce cc 1-74 of a master deck and sequence-punch numbers in cc 76=80 starting with 00010 and increasing by increments of 10. The second program will reproduce cc 7-80 and sequence-punch numbers in cc 1-6. The third program will reproduce cc 7-72 of the master deck, gang-punch constant identifying information into cc 73-80, and sequence-punch numbers into cc 1-6, starting with 000010 and increasing by increments of 10. The first program is good for sequencing or resequencing assembler language source programs. The latter two will do the same for COBOL or similar language source programs. The incrementing factor as well as the starting point of selected colums can easily be changed by the user. Programs are in machine language for any System/360 with a reader and punch physically addressed as 00C and 00D respecitvely.

Ordering Procedure: Order File Number 360D-00.0.009.

Distribution will be in card form only.

S/360 COS-VERSION 5, LEVEL 0 COMPATIBILITY OPERATION SYSTEM (REPLACES VERSION 4). This release of the Compatibility Operating System incorporates many corrections to the existing version as well as additional support and performance enhancements in certain areas. Users of earlier versions of the Compatibility Systems should pay particular attention to the following items which are implemented under this release in a different manner.

1. Include support for the simulation of the 1400 card reader, card punch, and printer on 2314 disk extents.

2. Includes support for the simulation of up to five (5) 1311's on 2314 disk storage. As many as four (4) 1311's may be simulated on a single 2314 module.

3. Includes support for the simulation of a 1301 module on two and one-half (2 1/2) modules of 2314 disk storage.

4. Includes support for the simulation of one or two modules of 1405 disk storage on 2314 disk storage. Each 1405 module on separate 2314 modules.

5. Includes support for the simulation of disk file scan operations utilizing the File Scan Feature (#4385). When the number of sectors to be scanned exceeds 20, a performance improvement will be realized over previous versions.

6. Includes support for selective 1405 disk verification on a drive for drive basis consistent with 1311 support.

7. Includes support for an additional (optional) 1400 control card parameter enabling the user to specify at execution time the instruction address of the normal 1400 end-of-job Halt instruction. Upon encountering this halt, control will be automatically transferred to DOS Job Control.

8. Includes an improved Program Mode Switch Supervisor Call interrupt handler.

9. Includes support for the simulation of 1401/1440/1460 Write and Space Suppress operation and 1401/1460 Write Word-Mark operation.

10. Includes an improved simulation technique for the simulation of 1400 carriage control operations.

11. Includes performance enhancements for the simulation of 1402 card reader stacker select operations.

12. Includes support for the simulation of 1401/1460 Read Column Binary operations with optional translation of the binary data in the normal card input area.

13. Includes support for the simulation of 1401/1460 Column Binary punch operations.

14. Includes performance enhancements for the simulation of 1402/1442 punch operations when the System/360 punch unit is a serial punch. This enhancement requires no 1400 program modification and improves performance by a factor of up to 2 1/2 depending upon the actual number of columns being punched.

15. Includes performance enhancements when the System/360 card reader is a 2501. Performance improvements of up to 60% can be realized.

16. Includes DOS Supervisor alterations to allow DASDFP to be specified when generating a DOS Supervisor.

17. Includes support for four user specified exits, automatically generating necessary linkage from the error routine, punch routine, printer routine, and end-of-job routine.

18. Includes ten additional parameters.

19. Includes support for Model DE (24K) systems.

20. Includes support for 144 print position 1443.

22. Includes support to write a tape mark at the beginning of each subsequent tape reel when the 1400 referenced card punch or printer are being simulated on magnetic tape. This conforms to standard DOS usage.

23. Includes support of the 1052 cancel key during the simulation of 1400 Console Read operations.

24. Includes test to insure that 1405 simulated disk files have been properly formatted.

This support is consistent with 1301-1311 support.

25. Includes support for altering quadrant assignments of 1311 disk drives when simulated on 2314 disk storage through normal operator service routines.

Ordering Procedure: Order File Number 360D-05.1.005.

The program material can be obtained on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

The requester must indicate whether a 9 track or 7 track DTR is required. If not specified, a 9 track DTR will be provided.

No tape submittal is required. The DTR will be provided by the Library.

S/360 HOUSTON GENERAL ONESTEP MONITOR. The objective of this program is to provide a generalized monitor program under OS/360 to allow compile, compile-linkedit, and compile-linkedit-go to be executed in only one job step. This will prove beneficial in speeding up these processes because the job scheduler need be brought in only once for each situation, resulting in less OS overhead.

Execution of the monitor is accomplished through normal JCL and it, in turn, causes invocation of the proper compiler, linkage editor, and the program to be tested by use of the LINK macro.

The program is patterned after the FORTRAN H ONESTEP MONITOR which is avilable in the IBM Program Library (360 D 03.2003). This program extends the capabilities of the above to include support for all compilers running under OS, as of Release 13, except for ALGOL.

This program can aid in speeding up the developmental activity by increasing throughput.

Ordering Procedure: Order File Number 360D-03.2.006.

The program material can be obtained in card form or on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

The requester must indicate whether a 9 track or 7 track DTR is required. If not specified, a 9 track DTR will be provided.

No tape submittal is required. The DTR will be provided by the Library.

S/360 PROCK - A PROGRAM CHECK ROUTINE IN MACRO FORMAT. PROCK is a DOS Assembler macro which generates a comprehensive program check routine for use with the DOS STXIT macro. When a program check occurs the job name and most of the old PSW is printed on the console typewriter. The operator has four options in replying:

1. Go to end of job without a core dump.
2. Go to end of job with a core dump.
3. Mask out decimal overflow and fixed point overflow and continue the problem program.
4. Mask out decimal overflow and fixed point overflow, dump the background area, and continue the problem program.

PROCK will run on any DOS configuration with a console typewriter and can be of significant aid in both debugging and production runs. It requires under 300 bytes of core.

Ordering Procedure: Order File Number 360D-01.4.008.

The program material can be obtained in card form or on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

The requester must indicate whether a 9 track or 7 track DTR is required. If not specified, a 9 track DTR will be provided.

No tape submittal is required. The DTR will be provided by the Library.

S/360 TESTMISP - A TESTING AID FOR INSTALLING MEDICAL INFORMATION SYSTEM PROGRAMS. TESTMISP is a System/360 DOS Program which allows an Assembler Language programmer to test application programs to be used in a hospital information system utilizing the Medical Information System Programs (MISP). A working knowledge of functions and requirements of MISP is a prerequisite for the use of TESTMISP. Information concerning MISP is available through your IBM Marketing Representative.

This program simulates MISP operation by performing the same functions, simulating these functions and/or printing a trace of the functions indicated in the application program.

Full core and disk storage capacity and the terminals required in an operational hospital information system are not required in order to utilize TESTMISP. If additional core storage is available, however, the user may continue to test programs in the background partition concurrent with "live" operation after MISP is installed.

Ordering Procedure: Order File Number 360D-04.4.007.

To obtain program meterial, submit one 2400 foot reel of magnetic tape. Specify whether 7 or 9 track recording is required. If not specified, 9 track recording will be used.

The required tape may be ordered from IBM or supplied with your request for the program.

S/360 OPERATING SYSTEM PL/I MACRO PREPROCESSOR. A program which accomplishes most of the replacement functions of the OS PL/I (Macro) processor stage, such as compile - time functions, declares, and assignment statements. It reads PL/I source text, scans for known macro identifiers, calls macro routines which replace the identifiers with generated text, and writes the modified program text on a data set which is used as input to either the PL/I processor or compilation stages.

Functions such as rescanning and compile - time loop expansion are not accomplished. The major program is written in PL/I, occupies 40K, and requires OS and two data sets.

Major advantages: Decrease compilation time by 30 to 55%; Ability to use complete PL/I language in compile - time functions; Provide spacing and line control for neater PL/I source listings.

Ordering Procedure: Order File Number 360D-03.6.005.

The program material can be obtained in card form or on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

The requester must indicate whether a 9 track or 7 track DTR is required. If not specified, a 9 track DTR will be provided.

No tape submittal is required. The DTR will be provided by the Library.

S/360 THRESH - A GEOMETRY PROGRAM FOR BUBBLE CHAMBER EVENTS. THRESH is a geometry program for bubble chamber events. For each event processed the program reconstructs the event in space, i.e., finds joint coordinates and track parameters. The program deals with a maximum of four stereoscopic views. It is assumed that the chamber is in a uniform magnetic field, perpendicular to the front glass and that the charged particles describe pure helices. A new set of routines (MATCH) performs the function of matching track images in three views.

The present program for the S/360 is an H-level FORTRAN IV version of the program dev-

eloped by CERN in Geneva, Switzerland (Version 12/06). Running under OS requires 256K bytes of core memory, and a recommended minimum of three disks (1 system, 1 scratch, 1 program), and two tapes.

Ordering Procedure: Order File Number 360D-17.2.008.

The program material can be obtained on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

The requester must indicate whether a 9 track or 7 track DTR is required. If not specified, a 9 track will be provided.

No tape submittal is required. The DTR will be provided by the Library.

IBM 1780 (GEOSPACE) PLOTTER SUBROUTINE FOR OS/360. An IBM System/360 FORTRAN-callable subroutine which permits the user to write data from main core to the IBM 1780 (Geospace) plotter independent of timing considerations, and to issue assorted control commands to that device. The program is written in Operating System Assembly language and requires a 131K byte machine, but can easily be modified to run in a 65K byte environment. The subroutine accepts data in 200 to 64000 byte blocks, normally from alternate buffers, and permits overlapped processing.

Ordering Procedure: Order File Number 360D-08.6.005.

The program material can be obtained in card form or on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

The requester must indicate whether a 9 track or 7 track DTR is required. If not specified, a 9 track DTR will be provided.

No tape submittal is required. The DTR will be provided by the Library.

CERN SUMX. A DATA SUMMARIZATION PROGRAM FOR THE IBM/360. CERN SUMX analyzes information about large number of events and produces histograms, scatter diagrams, lists and ordered lists. Facilities are provided to select subsets of events according to criteria defined on control-cards, and to allow the user to add routines for computing event by event quantities not immediately available. The program was originally written at Berkeley, but the present version was completely rewritten at CERN in 1965-1966. (SUMX 466, version 5.25). The language is FORTRAN IV, level H for S/360. CERN SUMX requires 256K of core memory without overlays, three tape drives and a recommended minimum of 3 disk drives.

Ordering Procedure: Order File Number 360D-17.2.006.

To obtain program material submit one 2400 foot reel of magnetic tape. Specify whether 7 or 9 track recording is required. If not specified, 9 track recording will be used.

The required tape may be ordered from IBM or supplied with your request for the program.

S/360 CERN GRIND - A PROGRAM FOR KINEMATIC ANALYSIS OF EVENTS IN BUBBLE CHAMBERS. GRIND is a kinematic fitting program written by CERN. This is an H-level FORTRAN IV program based on the 03/05 CERN version dated June 1966.

GRIND will process the geometry of each event (THRESH output) as well as the experiment dependent TITLES containing various parameters, mass hypothesis to be tested, convergence cirteria, etc.

GRIND can produce an output tape for further processing on SLICE as well as punched cards identifying tenable hypothesis.

This program under OS requires 256K of core, and a recommended minimum of three disks (1 system, 1 scratch, 1 program) and two tapes.

Ordering Procedure: Order File Number 360D-17.2.007.

The program material can be obtained on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

The requester must indicate whether a 9 track or 7 track DTR is required. If not specified, a 9 track DTR will be provided.

No tape submittal is required. The DTR will be provided by the Library.

1130 IMPROVED HYPHENATION PACKAGE FOR 1130 TCP. Increases in speed of up to 1500 30-character lines per hour can be realized with this Improved Hyphenation Package for the 1130 TCP. A single compressed dictionary, based on word roots wherever possible, replaces the 1130 TCP separate dictionaries for suffix exceptions and probability exceptions. The package consists of the following programs: HTSTR, HLODR, Overlay 5, Overlay 7, HPREP. In addition, a set of about 3200 exception words is supplied.

Two new utility programs replace HYUP, namely HTSTR and HLODR. HTSTR accepts input from cards or from the keyboard and indicates the results of hyphenation on the console printer and/or by stacker selecting exception cards.

HLODR accepts input from the keyboard, from cards, or from 8-channel paper tape and loads the dictionary file accordingly. The new overlays 5 and 7 replace the corresponding overlays as distributed with 1130-DP-04X, Version 1, Level 2. HPREP, as before, initializes the dictionary but also initializes a dictionary index within Overlay 5. In addition to providing a more convenient method of testing and loading exception words, a significant savings in time can be gained by using this set of programs. Also, about 60 cylinders of disk storage space is freed for other purposes.

Ordering Procedure: Order File Number 1130-06.6.009.

Distribution will be in card form only for both basic and optional program material.

1800 CARD ASSEMBLER ADDRESS CONVERSION. This program produces a listing of an assembled 1800 list deck and indicates the corresponding instruction addresses for the core image and relocatable loaders. The addresses are printed in hexadecimal format. The program runs on any 1800 which has a 1442 Card Read Punch. It was written in Assembler Language.

Considerable time can be saved when debugging by referring to a listing produced by this program, rather than having to do hexadecimal arithmetic to convert every significant address. It can also be used to quickly find the program location of WAIT instructions.

Ordering Procedure: Order File Number 1800-04.0.001.

Distribution will be in card form only.

IBM Installation Newsletter

April 19, 1968

Issue No. 68-07

CONTENTS

(G means Green Section)

Distribution: Branch Offices - DP Management, Salesmen, Systems Engineers, FE Managers. Regions, Districts, Education Centers, Field Systems Centers, Federal Systems Centers, FE Area Offices, DPD HQ, FED HQ, WTC.

*Requires Immediate Attention

For IBM Internal Use Only

DPD HQ, White Plains, N.Y.

SELECTED SHORT SUBJECTS

The purpose of Selected Short Subjects is to bring together and highlight concise, factual and timely information which will indicate that action is to be taken by the IBM representative whose accounts are affected.

1. Solving Installation Problems

This is just a reminder of some of the sources available for helping you to solve installation problems.

In addition to other Branch Office, District, Regional and DPD HQ support functions, there is SECOM, RETAIN (FE), QUESTER, KWIC Index to TIE, Catalogs of Programs including Type III's, and the Installation Newsletter. The Field Systems Center, Installation Centers and DACS Centers are points of contact for problem solutions. Be certain that you are familiar with these resources. Use them before attempting to solve a problem which might have already been solved.

2. OS/360 BTAM BSC Auto Dial Feature Requirement

Current documentation does not indicate the prerequisite requirement for the Auto Dial feature in order to use BSC dial support. Sales Manual and SRL (TNL) Documentation will be forth coming which will provide the necessary information for obtaining this feature.

3. Teleprocessing - ACU 801A6 Power Off

Many IBM customers routinely turn off the power on their CPU equipment over night or on weekends. The data set and data auxiliary equipment, supplied by the common carrier, may be left with power ON during such periods. Normally this causes no problems and is an acceptable mode of operation.

However, the Automatic Calling Units 801A5 (which is not recommended for use with IBM equipment) and the 801A6 can not presently operate this way, without placing an erroneous OFF HOOK indication on the telephone, or TWX line, and ultimately causing alarms at the telephone company central office. A possible fix is being considered. Meanwhile, it is necessary that all 801A5 and 801A6 units have power disconnected when the associated IBM equipment will have power off for more than 15-30 minutes.

It should be noted that this is definitely a temporary solution and applies only to the 801A5 and 801A6. The earlier 801A1 and all 801C units are not affected.

OS/360 RELEASE 14 CHANGE NON-STANDARD CHARACTERS IN JCL

The following article concerns a change, effective with Release 14 of OS/360, which made non-standard characters unacceptable in the JCL DSNAME field, and an alteration which the user can make to obtain a temporary bypass until he can make the necessary changes to his JCL.

Special characters in the JCL DSNAME field are not acceptable in OS/360 Release 14. This is because the Reader/Interpreter has been replaced by a new version to be compatible with MVT. Some users have installed earlier releases with special characters such as in nnn-nn and nnn. nn. Those customers must ultimately conform to the new standard to assist in transition.

The following Superzap for module IEFVFA will cause the Reader/Interpreter to accept a non-standard character in the DSNAME field. It also permits any non-standard character to appear anywhere in JCL, which is an exposure that each user must anticipate.

```
VERIFY    061C 4120,  0017
REP       061C 47F0,  B5C4
```

The instruction appears at sequence number 17160000 of the Release 14 listing of module IEFVFA.

```
Before REP      LA  R2, ER MES8
After REP       B   FB3
```

Note that this is only a temporary bypass for Release 14. No permanent change will be made to the new scheduler to accept non-standard characters.

OS/360 - REVIEW OF ACCOUNTING FACILITIES

The following article includes information which is provided for planning purposes only. The actual availability, if any, of all such items and their specifications will be as provided through the official IBM procedures; the IBM Program Announcement letter and formal documentation. You are reminded that information in the "IBM Internal Use Section" is NOT to be shown to customers, or anyone else outside of IBM.

1. INTRODUCTION

The ability to account for and thereby charge for the time that a problem program uses the computing system has been the object of effort on the part of the users of OS/360.

This article summarizes the current capabilities of OS/360 in its different configurations; it highlights some areas of difficulty and presents some intermin solutions. The article does not cover all of the many modificatiors or extensions that have been made and therefore is not a definitive coverage of the subject.

The releases of the Operating System covered in this article are through Release 14. There is a section at the end devoted to planning.

For details on accounting routine writing, please refer to the System Programmers Guide (C28-6550).

2. MVT

With Releases 12 - 14, the MVT user has the option of selecting (with the SUPRVSOR macro) Job Step Timing as a SYSGEN option. This will cause instructions to be included in the Task Dispatcher which have the effect of accumulating "TASK" time for an Initiator Task whenever any lower priority task is running, (its job step task and others) or when the hardware WAIT STATE is activated. This is intended to accumulate CPU active time plus WAIT time for each job step. The technique allows the Operating System to terminate a Job Step if its run time exceeds a specified time interval. The user can control this interval with the TIME=parameter on the EXEC Statement, or the Reader/Interpreter can supply a default value.

A user's accounting routine will be entered at Step Termination and also at Job Termination. A parameter list pointer is supplied in Register 1 when the accounting routine is entered. At offset 12 from the address in Register 1 there is a pointer to a three byte field which contains the Job Running time (the accumulated step-times). At offset 20 from Register 1 there is a pointer to a three byte field which will contain the time that the current step has accumulated. These two times are available to the accounting routine at Job Termination and Step Termination. The accounting routine can determine whether a Step or Job is terminating by testing the Step-name Pointer (off-set 4 from Register 1) for zero. A zero condition indicates Job Termination. It should be noted that Register 0 will not contain a valid entrance code as it does with PCP or MFT-I. Register 12 contains the address of a system table (the Linkage Control Table) which may be used to write information to a SYSOUT queue for later processing by an Output Writer.

The Step and Job time data fields will be zero if Job Step Timing was not generated into the System, however, the pointers to the timing data fields will be intact.

Some users find significant variations in the time recorded for a job or step from one run to the next. In an attempt to correct this variation, considerable time has been spent isolating the sources of variation to the time. The following list represents the major causes of inaccuracy.

a) The current algorithm for accumulating time makes it possible for a task to execute without being billed for all of its WAIT time. This would be the case when a task of priority X

performed an I/O operation and WAIT's for the completion of that operation, and during that WAIT a higher priority task became ready and was given control. The timer for the WAIT'ing task will be removed from the timer queue, with the result that the WAIT'ing task will not be charged for all of the time that it WAIT'ed.

b) The resolution of the standard Interval Timer (the hardware timer) is approximately 16.6 ms.

Measurements have indicated that a task may receive control, execute a number of instructions and lose control to another task, all within the 16.6 ms. This will make it appear that the task used no time. On the other hand, a task may receive control just as the Interval Timer decrements; in this case the task could appear to have used more time than it actually used. The faster the CPU, the more inequitable this charging becomes.

c) When data is being transferred through a channel, the CPU may be delayed from referencing main memory to allow the channel to fetch or store the data. During this time the Interval Timer continues to run.

d) There is a known problem in the Task Dispatcher (see page 3, Internal Section, Newsletter No. 68-01, 'OS/360 MVT Job Step Timing Error Correction'). This problem causes the dispatcher to either leave tasks on the timer queue when they should be removed, or to not place them on the queue when they should be on. This can be caused in two ways.

1. The user supplies a Job Step which asks for 1440 minutes of time (no Job Step timing).

2. An Output Writer task has attached a data-set writer task.

Under these circumstances the dispatcher suspends enqueueing or dequeueing the timer for all initiator tasks of lower priority than the task without a timer.

e) While a task (A) is executing instructions, an I/O interrupt causes control to be given to IOS, the timer for task (A) remains on the timer queue until IOS has processed the interrupt and has given control to the dispatcher. It is possible that the cause of the interrupt is an Input/Output operation initiated by task (B). In this case (A) is charged for (B's) interrupt.

f) With the exception of point 2d above, no single one of these conditions causes large variations. However, the cumulative effect of all of them does cause variations from run to run and from one CPU model to another.

There are interim solutions for some of these problems, for others there are none at this time.

3. INTERIM SOLUTIONS:

Two solutions are available for the Dispatcher problem.

a. That referred to in Installation Newsletter 68-01 which corrects the problem in the Dispatcher, but does not affect the truncation of WAIT time mentioned in 2a above.

b. An alteration has been developed to the Dispatcher which allows the accumulation of CPU active time only. See "OS/360 MVT CPU Timing Alteration" in issue 68-06 of the Newsletter. This eliminates the "truncation of WAIT time" and also eliminates the problem described above. It produces times which are more consistent, but which do not reflect the total activity of the problem program.

By eliminating the accumulation of WAIT time, a program that comes into the system and WAIT's endlessly will not be terminated by Job Step Timing. Only the program that executes for too long will be terminated.

A caution should be observed; the time that is to be allowed will not be the same (numerically) that

the program would have executed with in PCP. It will be dependent upon the nature of the program (its I/O to CPU ratio).

Another consideration is the modification of System code. This imposes upon the modifier (user) the responsibility for reviewing the code with each revision to the system.

The Dispatcher will be subject to changes in subsequent releases of the system to provide improvements to the present code.

c) The accounting routine data set writer (IEFWAD) can be included in the MVT or MFT-II Job Scheduler to provide these systems with a function which is similar to PCP. (See Exhibit 1 for instructions for including IEFWAD.)

In the PCP configuration, the operator has the opportunity to enter the location of the SYS1. ACCT data set with the initial SET command. (eg. SET DATE= xx. xxx, ACCT= xxx). This capability is a function of the initialization of the system, and not of the Job Scheduler. Therefore, the simple inclusion of the IEFWAD module in the scheduler will not provide that capability. IEFWAD does, however, allow the operator to change the location of the SYS1. ACCT data set when the data set fills up, or cannot be found (message IEF506D SYS1. ACCT, xxx(, N) NOT PRE-ALLOCATED). The operator need not indicate the data set location until the first use of the data set, at which time he may direct the data set writer to any device that contains the data set.

In MVT and MFT-II a Reader/ Interpreter task is started from a cataloged procedure. This procedure may direct the Reader/ Interpreter to a data set which contains Job Control Language and serves as the job stream to this reader. A procedure may be added to the procedure library which will cause the scheduling of a job to print, punch, or otherwise process the accounting data contained in SYS1. ACCT. (See Exhibit 2 for an example of how to do this.) The operator need only enter START DRAWOFF and the job will be scheduled and executed.

4. MFT

Because MFT systems do not have a separate task for the initiator, but rather XCTL from initiator to problem program, there is no TCB to set a timer for without limiting the problem program's use of the timer functions. As a result of this, it has been the practice to obtain the starting and ending "time-of-day" and use these figures to calculate elapsed step-time.

In MFT, there has been some inaccuracy because of higher priority partitions interfering with the batched partition, but it has not been indicated to be a significant problem.

In MFT - II however, multiple batch partitions may be run with system Reader and Writer tasks. The problem now becomes very much the same as the MVT problem; how to charge one job in an equitable manner when there are many such jobs in the system at the same time.

The user who is planning to to to MFT-II with the initial release must be made aware that the only timing facility provided is the time of day at initiation and termination.

This is recognized to be a limitation and every effort is being made to improve it.

5. OUTPUT FROM ACCOUNTING PROGRAMS:

a) In all configurations of OS/360 (PCP, MFT-I, MFT-II and MVT) the user-supplied accounting program may use the WTO SVC to output accounting information on the console device.

b) In all configurations of OS/360, the accounting program may write messages to the SYSOUT data-set.

c) In the PCP or MFT-I configuration, the users accounting program may write into an accounting data-set named SYS1.ACCT.

d) In an MVT configuration, the user's accounting routine may write into the system LOG data set.

e) In the initial release of the MFT-II configuration, there will be no support of SYS1.ACCT or the LOG. This is recognized to be a limitation and work is in process to improve it. The implementation will probably be a function similar to the LOG of MVT. (See Interim Solutions, paragraph c.)

6. OTHER CONSIDERATIONS

The sample accounting routine in SAMPLIB uses the facility of the SYS1.ACCT data set. For this reason, the program cannot be used in MVT or MFT-II.

7. PLANNING INFORMATION

The following planning information is provided in the event that changes to a user's system are being considered. Note that the first paragraph in this article applies to the following:

a. The next release of OS/360 MVT may include an Accounting Exit at initiation time, the entry code in Register 0 and a pointer to the step-number and job-fail bytes (as PCP now has). Initial documentation may be misleading on this point.

b. It is presently planned that, in a future release, MFT II will be updated to have Step Timing accounting facilities. This will be done as soon as possible.

c. More extensive accounting facilities may be included in future releases of OS/360. These facilities would provide additional information and services and enable installations to monitor the use of their systems.

d. A more consistent means of accumulating time (similar to the CPU Timing Alteration referred to above) is being considered for MVT, including the ability to break an endless WAIT condition.

OS/360-REVIEW OF ACCOUNTING FACILITIES

```
              INCLUDING IEFWAD IN THE JOB SCHEDULER WITH A USER
ACCOUNTING ROUTINE.
THE IEFWAD PROGRAM CAN BE FOUND IN THE MODLIB DATA SET OF
YOUR STARTER SYSTEM.
THE SYS1.ACCT DATA SET SHOULD BE ALLOCATED IN ACCORDANCE WITH THE
RULES IN THE  SYSTEM PROGRAMMERS GUIDE. (VIZ. NO SECONDARY SPACE
ALLOCATION,HAVE CONTIGUOUS SPACE AND NOT BE CATALOGUED.)
*
//ADDWAD        JOB     0,        ,MSGLEVEL=1
//STEP1         EXEC    PGM=IEWL,PARM='XREF,LIST,LET,NCAL',          ,
//              REGION=96K
//SYSPRINT      DD      SYSOUT=A
//SYSLMOD       DD      DSNAME=SYS1.LINKLIB,DISP=OLD
//SYSUT1        DD      UNIT=2311,SPACE=(CYL,(4,2)),DISP=NEW
//ACCOUNT         DD      DSNAME=SYS1.ACCT,UNIT=2311,                  ,
//              VOLUME=REF=SYS1.SVCLIB,DISP=(NEW,KEEP),              ,
//              SPACE=(121,(400),,CONTIG)
//MODLIB        DD      DSNAME=SYS1.MODLIB,UNIT=2311,DISP=OLD,       ,
//              VOLUME=SER=DLIB03
//SYSLIN        DD      *
        OBJECT MODULE FOR IEFACTRT
         INCLUDE        MODLIB(IEFWAD)
         INCLUDE        SYSLMOD(MODULE NAME)
         ALIAS          MODULE ALIASES
         ENTRY          MODULE ENTRY POINT
         NAME           MODULE NAME(R)
/*
```

Exhibit 1

OS/360 - REVIEW OF ACCOUNTING FACILITIES

```
        READER/INTERPRETER PROCEDURE TO PRINT SYS1.ACCT.
THIS PROCEDURE MAY BE INSERTED INTO THE PROCEDURE LIBRARY OF
AN MFT-II OR MVT SYSTEM.
THE OPERATOR MAY THEN ISSUE A 'START DRAWOFF' COMMAND,WHICH WILL
CAUSE THE SCHEDULING OF THE CATALOGUED PROCEDURE TO PROCESS THE
SYS1.ACCT DATASET.
*
//ADDPROC       JOB     0,       ,MSGLEVEL=1  ADD READER PROCEDURE
//STEP1         EXEC    PGM=IEBUPDTE,PARM=NEW
//SYSUT2        DD      DSNAME=SYS1.PROCLIB,DISP=OLD
//SYSPRINT      DD      SYSOUT=A
//SYSIN         DD      DATA
./        ADD         NAME=DRAWOFF
./        NUMBER      NEW1=0,INCR=25
//IEFPROC       EXEC    PGM=IEFIRC,                                      ,
//              REGION=52K,              READER BASIC REGION            X00040000
//              PARM '80103005001024905210D11A     '  DEFAULT OPTIONS   ,00060000
//              ******BPPTTTOOOMMMIIICCCRLSSSSSSSS  PARM FIELDS DEFINED ,00080000
//              PROGRAMMER NAME AND                          B           ,00100000
//              ACCOUNT NUMBER      NOT NEEDED                           ,00120000
//              PRIORITY 01                                  PP          ,00140000
//              JOB STEP INTERVAL 30 MINUTES                 TTT         ,00160000
//              PRIMARY SYSOUT SPACE 50 TRACKS               OOO         ,00180000
//              SECONDARY SYSOUT SPACE 10 TRACKS             MMM         ,00200000
//              READER/INTERPRETER DISPATCHING PRIORITY 249  III         ,00220000
//              JOB STEP DEFAULT REGION 50K                  CCC         ,00240000
//              DISPLAY AND EXECUTE COMMANDS 1               R           ,00260000
//              BYPASS LABEL OPTION 0                        L           ,00270000
//              SYSOUT UNIT NAME D11A                        SSSSSSSS     00280000
//IEFRDER   DD   DSNAME=SYS1.PROCLIB(DRAW),DISP=OLD
//IEFDATA       DD      DUMMY
//IEFPDSI       DD  DSNAME=SYS1.PROCLIB,          PROCEDURE LIBRARY      ,00380000
//              DISP=OLD                                                  00400000
./        ADD         NAME=DRAW
./        NUMBER      NEW1=0,INCR=24
//DSKPRT JOB MSGLEVEL=1
//STEP EXEC PGM=IEBPTPCH
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD UNIT=2311,DSNAME=SYS1.ACCT,DISP=OLD,DCB=(BLKSIZE=121)
//SYSUT2 DD SYSOUT=A
//SYSIN   DD  DSNAME=SYS1.PROCLIB(DR),DISP=OLD
./        ADD         NAME=DR
./        NUMBER   NEW1=0,INCR=25
 PRINT MAXFLDS=1
 RECORD FIELD=(120,1,,1)
./        ENDUP
/*
```

Exhibit 2

OS/360 FUTURE RELEASE PLANNING INFORMATION

The following information is provided for planning purposes only. The actual availability and specifications of all items will be as announced through the official IBM procedures, the Program Announcement and formal documentation.

Installation Newsletter 68-03, page 2, contained planning information for OS/360 Release 15. The plan for this release has been altered as a result of analyzing user requests for fewer full releases of OS/360, to minimize the cost of upgrading, and to provide for a longer maintenance period for MFT- I. The decision has been made to:

1. Distribute a Release 14 CMR (Compiler Maintenance Release), rather than Release 15 as previously planned .

2. Instead of the original Release 15, provide a combined Release 15/16. The planning information previously published (issue 68-03) for Release 15 will be applicable to Release 15/16. Additional planning information about Release 15/16 will be made available in a future issue of the Installation Newsletter.

Plans are that Release 14 CMR will include the COBOL and FORTRAN Compilers with their related libraries, and possibly PL/I. These components can be incorporated into Release 14 through a processsor--only SYSGEN. It is expected that users who have already ordered Release 14 and use COBOL and/or FORTRAN will order Release 14 CMR. New users will require both Release 14 and the Release 14 CMR.

The Release 14 CMR will provide fixes for over 100 APAR's. It is possible that Release 14 CMR may be available as early as May, 1968.

DOS/360 RELEASE 16 OPTIONAL MATERIAL INITIALIZE DISK ERROR

DOS/360 Release 16 (Program Announcement P68-53) Optional Program Material tapes contain an initialize disk program which does not properly flag bad tracks. The stand-alone BPS program or the Initialize Disk program at the beginning of Release 16 distribution tapes should be used if pack initialization is required before restoring the optional program material tapes to disk.

DOS/360 VERSION 3 SERVICE AID PLANNING INFORMATION

The following information is provided for planning purposes only. The actual availability and specifications will be as announced through the official IBM procedures, the Program Announcement letter and formal documentation.

The initial release (P68-53) of DOS/360 Version III, (DOS Release 16), contains the first of five service aids planned for use with Version III. This aid is I/O Interrupt Trace. Four other aids are presently under consideration.

I/O Interrupt Trace is designed to allow I/O activity of programs run under DOS/360 to be traced. Tracing consists of recording the I/O old PSW and the CSW when an I/O interruption occurs, or recording the Device Address, the CCD address and the CSW when the CSW is stored in response to an SIO instruction issued by the DOS supervisor. After recording, the data may be preserved in a rotating buffer in core (first entry overwritten when the area is full, etc.), or output on a printer, punch or tape unit. When Punch or Tape output is used, the CEAIDLST utility program is required to provide readable output data.

1. Prerequisites

   a) Program Requirements

      I/O trace requires a DOS Supervisor, (Version 3 or later) in which the FOPT macro option 'CE=YES' or 'CE=nnnnn' has been exercised. ($10,240 \geq$ nnnnn ≥ 600)

      The following Phases must be present in the CORE IMAGE LIBRARY:

      CEAID
      CEAID001
      CEAID002
      CEAIDLST
      $$BZCE01

   b) Equipment Requirements

      Minimum DOS/360 requirements, plus 32K bytes of main storage. When tape, printer, or punch output is used, the output device must be dedicated to the trace program, and should not share a Selector Channel or a control unit with other programs being executed.

      Complete information on operation, options, etc. is contained in form number Z24-5091 (IBM S/360 DOS Version III C. E. Serviceability

Programs). This publication will be updated as the additional aids are released.

It is imperative that the customer exercise the CE option, (CE=YES or nnnnn), when assembling his DOS Version III Supervisor in order to obtain the benefits of these aids.

If additional information is required, contact your FE Branch Office.

S/360-20 NEW CATALOG OF PROGRAMS

A new Catalog of Programs for IBM System/360 Model 20 (Form Number C20-1691) will be made available and announced in the Publications Release Letter (PRL).

The new catalog may be ordered from the IBM Distribution Center, Mechanicsburg following announcement of availability in a PRL. This new Catalog of Programs will contain abstracts and ordering information for all programs available to users of System/360 Model 20 from the Program Information Department.

All System/360 Model 20 programs which appeared in the Catalog of Programs for IBM System/360 (Form Number C20-1619-4) have been deleted.

TELEPROCESSING - TELPAK TARIFF REVISIONS

Prior to using the following information, be sure that you have reviewed "Planning a Teleprocessing System, page GI-3, and competition page GI-1, in the Sales Manual and understand the IBM policies which apply.

AT & T has filed new revisions in its Series 5000 channels - Base Capacity (formerly known as Telpak) which will increase the cost of these facilities in several ways on June 1, 1968:

1. Mileage charges are increased.
2. Service terminal charges are increased.
3. Telegraph and subvoice channel capacity is reduced.

Exhibit 1 shows a comparison of the present and future rates where changes will occur. Notice that the high speed type numbers are also changing. In addition, the old 5752 and 5801 types which ran at 105 and 500 kbps, respectively, are to be removed.

The reaction of users to these changes indicates that the effective date may be postponed, possibly until 3Q68. However, if you have customers with Telpak installed, or proposals outstanding which use Telpak, you should be considering the impact of this change.

Two alternatives, which definitely can look better under the new rates than under the old, are the IBM Remote Multiplexors (2712 and 2905) and the Shared Line Adapter modem. Since telegraph channel users will be impacted the most by this revision, you may well find that upgrading to fewer channels of IBM terminals operating at 134.5 bps and using the Shared Line Adapter will compare favorably with telegraph systems.

1600 BPI PHASE ENCODED TAPE R/W AND CAPACITY

When calculating the capacity and Read/Write times for 1600 BPI tapes, 82 bytes must be added to each tape block.

Forty-one bytes are added to the beginning and 41 bytes to the end of each block, automatically when the tape is written, to obtain synchronization for forward and backward tape read operations.

These added bytes will have the most pronounced effect on tape operations when small blocks are used.

TELEPROCESSING - TELPAK TARIFF REVISIONS

NOW

$25/mile	Base Capacity 5700 (60 equiv Voice Channels)
$45/mile	Base Capacity 5800 (240 equiv Voice Channels)

Service Terminals***

First/Exchange	Each Add'l	
$15	5	5102 (75 bps Telegraph) up to 12 per equiv Voice Channel
15	5	**5106 (150 bps Subvoice) up to 4 per equiv Voice Channel
15	5	5302 (Voice grade data)
425	375	*5705 (19,200 bps data) uses 6 equiv Voice Channels
250	195	*5702 (40,800 bps data uses 12 equiv Voice Channels
364	-	*5704 (50,000 bps data) uses 12 equiv Voice Channels
455	-	*5754 (230,400 bps data) uses 60 equiv Voice Channels

FILED EFFECTIVE JUNE 1, 1968
(Items changed from above chart are underlined)

$28/mile	Base Capacity 5700
$60/mile	Base Capacity 5800

First/Exchange	Each Add'l	
$25	10	5102 (75 bps Telegraph) up to 6 per equiv Voice Channel
25	10	**5106 (150 bps Subvoice) up to 3 per equiv Voice Channel
25	10	5302 (Voice grade data)
425	375	*5703 (19,200 bps data) uses 6 equiv Voice Channels
425	300	*5701 (40,800 bps data) uses 12 equiv Voice Channels
425	300	*5701 (50,000 bps data) uses 12 equiv Voice Channels
650	-	*5751 (230,400 bps data) uses 60 equiv Voice Channels

*Each of these service terminal charges includes a data set or data station.
**An additional $25 station arrangement is required at each premise served by this channel.
***One service terminal charge is incurred for each connection to the line, even when more than one is on the same premises.

Exhibit 1

SECOM ON MICROFICHE TO BE DISCONTINUED

Because of costs and an apparent lack of use of SECOM on the CE microfiche at customer locations, this method of distribution will be terminated as of May 1, 1968. Other methods of distribution will be continued--i. e., daily broadcast, and weekly abstracts mailed to branch offices. Retrieval of detail information can be accomplished by ITPS, dial-up 1050 and the leased line 2740 terminals being installed by CE in all of their branches during 1968. 2740's will also be installed in some DP support locations (i. e. Field Systems Centers).

If this decision causes a problem in availability of SECOM information, consideration may be

given to a SECOM, microfiche package available (with full text) each week by subscription. The need for this type of service or any other comments on SECOM should be sent to:

DPD SECOM Coordinator
Department 713
DPD HQ

Direct inquiries concerning the contents of this Newsletter to:

W. P. Gaffigan, Editor
IBM Installation Newsletter
Installation Planning-#713
DPD HQ
112 East Post Road
White Plains, New York 10601
Telephone 914 WH9-1900 Extension 4781

INFORMATION ABOUT THE NEWSLETTER

EVERY SALESMAN AND SYSTEMS ENGINEER SHOULD RECEIVE ONE COPY OF THE NEWSLETTER. Each FE Branch Manager should receive five copies for distribution to Customer Engineers.

The IBM Installation Newsletter is distributed only to IBM locations and is not available to customers. The Newsletter is separated into two parts. The first part contains information which is intended only for IBM personnel and the pages are labeled For IBM Internal Use Only. THESE PAGES, OR REPRODUCTIONS OF THEM, ARE NOT TO BE GIVEN TO CUSTOMERS. However, using your own good judgement, you may discuss pertinent information from these pages with a customer. The second part contains information which may be REPRODUCED by the Branch Office at their discretion and given to customers. These pages are not labeled with any restrictive classification. Note that when any abstracts for Type III programs are given to the customer, he must be informed that the disclaimer for Type III programs as contained in the Newsletter, applies. YOUR COOPERATION IN PROPERLY HANDLING THE CONTENT OF THE IBM INSTALLATION NEWSLETTER IS NECESSARY FOR ITS CONTINUED PUBLICATION.

Changes to the quantity of the Newsletter shipped are to be made by the Administration Manager. (NOTE THAT EXTRA COPIES OF THE NEWSLETTER ARE NOT TO BE ORDERED FOR DISTRIBUTION OF INFORMATION TO CUSTOMERS.) He should update and send the DP Administration Managers' Communication Control card (form number N44-780) to DP Communication Control, DPD HQ.

NEW PUBLICATIONS ANNOUNCED BY PRL'S

The weekly PRL's (Publications Release Letters) are used to insure that all Salesmen and Systems Engineers are aware of new or revised Marketing Publications. Normally, each issue of the Newsletter will contain information from two PRL's, one following the other. The information will be placed in the Newsletter in its original form with no rearrangement of form numbers or titles. It is not intended to replace existing information and distribution sources. You should be certain that you are aware of these sources.

PRL #11 March 15, 1968

REFERENCE SOURCES

FORM NO.	TITLE	DISPOSITION
N20-7070.33	7070 SRL Newsletter Re: A22-6699-4	Scrap N20-7070.32

MARKETING PUBLICATIONS

320-3201-0	Control Systems Analysis Program II (CSAP II)	NEW
320-3203-0	Possible Performance Improvements by Microprogramming for System/360 Models 40, 50 & 65	NEW
320-3207-0	Extended Use of the Coulomb Approximation: Mean Powers of r, a Sum Rule, and Improved Transition Integrals	NEW
320-3208-0	Analog, Digital, and Hybrid Computers in Process Control	NEW
320-3209-0	Line Profiles in the Upper Atmosphere: The Voight Function	NEW
320-3215-0	EXCPAM (Execute Channel Program Access Method Under OS/360)	NEW
320-3216-0	Observations on the Design of High-Performance Digital Computers	NEW
320-3218-0	A Reexamination of Homogeneous Nucleation Theory: Statistical Thermodynamics Aspects	NEW
320-3219-0	Time-Shared Critical Path Calculations	NEW
320-3220-0	Asymptotic Behavior of Certain Nonlinear Boundary Value Problems	NEW
320-3221-0	Data and Systems Requirements in Meteorology	NEW
320-3222-0	Numerical Algorithms for the Solution of the Evaporation of a Raindrop	NEW
320-3223-0	The Advantages of Using the IBM System/360 Computer for Large-Scale Scientific Problems	NEW
320-3224-0	Free Energy of Formation of Droplets from Vapor and Dependence of Surface Tension on Radius	NEW
320-3225-0	Conjectures Concerning the Connection Between Stability in the Small of Autonomous Differential Equations and the Roots of the Jacobian Matrix	NEW
320-3226-0	Theory of the Diffusivity Factor for Atmospheric Radiation	NEW
320-3227-0	Predicting the Critical Supersaturation for Homogeneous Nucleation of Vapor Condensation	NEW
320-3228-0	Study of the Mammalian Adrenal Glucocorticoid System by Computer Simulation	NEW
320-3231-0	A Subroutine for the Evaluation of the Exponential Integral with Fifteen Significant Figure Accuracy	NEW
520-1410-1	You get more than machines from IBM	Scrap 520-1410-0
549-0055-2	Select Composer Instruction Manual Supplement	Use 549-0055-2
549-0202-0	Composer Type Style Booklet	NEW
A24-3125-3	1050 Operator's Guide	Use A24-3125-2, N27-3000 & N27-3014
A24-3493-0	IBM 1450 System Summary	NEW
A24-3516-0	2938 Model 1 and Model 2 Array Processor RPQ Installation Manual - Physical Planning	NEW
C26-3605-3	S/360 Mod 20 Sterling Routines	Scrap C24-3605-2
C26-3718-1	IBM 1800 Multiprogramming Executive Operating System Introduction	Scrap C26-3718-0
C28-6329-4	IBM 7040/7044 FORTRAN IV Language	Use C28-6329-3, N28-0540 & N28-0551
F20-0011-0	Part-Period Algorithm - A New Economic Lot Sizing Techniques	NEW
J20-0002-1	Fundamentals of using QUIKTRAN	Scrap J20-0002-0
K20-0222-0	Audio Response for Application Control at Equitable Life Assurance Society of the United States	NEW
L27-3010-0	2905 Remote Multiplexer (Physical Planning)	NEW
N20-1070	Technical Newsletter for 1130 Mechanism Design System - Gears and Springs (1130-EM-01X) User's Manual Re: H20-0365-0	NEW
N22-0277	Changes to IBM S/360 Model 50 Functional Characteristics Re: A22-6898-1	NEW
N24-0400	TNL to IBM 1440 Instruction & Timing Summary Re: A24-3094-3	NEW
N24-0404	TNL to IBM 1440 DPS Physical Planning Installation Manual Re: A24-3007-5	NEW
N24-0414	TNL to S/360 Mod 30 Functional Characteristics Re: A24-3231-4	NEW
N27-2919	IBM S/360 Model 65 Functional Characteristics TNL Re: A22-6884-2 with N27-2909 & N27-2916	NEW
N30-2508	TNL to S/360 Operating System Basic Telecommunications Access Method Re: C30-2004-1	NEW
N30-5507	TNL to S/360 Disk Operating System Basic Telecommunications Access Method Re: C30-5001-4 with N30-5016, N30-5024 & N30-5505	NEW
N33-8001	TNL IBM S/360 Operating System ALGOL Language Re: C28-6615-1	NEW
N33-9025	TNL to Mod 20 DPS/TPS RPG SRL Re: C24-9001-3 with N33-9018	NEW
R20-4096-1	System/360 Operating System Language Interface Student Materials	NEW
R20-4097-0	IBM S/360 Computer Systems Simulator/360 (CSS/360) Education Guide	Scrap R20-4096-0
R20-4105-0	System/360 Operating System BTAM Coding Education Guide	NEW
R20-9100-3	IBM S/360 PL/I Coding Course Description	NEW
R20-9218-0	S/360 DOS/TOS Operation Course Description	Use R20-9100-2 NEW

PRL #11 (Continued)

R20-9227-0	System/360 Computer Simulator/360 Course Description	
R20-9233-0	IBM S/360 Operating System BTAM Coding Course Description	NEW
R29-0092-2	IBM S/360 Mod 20 RPG (MFCM) Text	NEW
R29-0167-1	IBM 519 Document Originating Machine Operation and Wiring Notebook	Use R29-0092-2
R29-0242-2	IBM Computing System Fundamentals Notebook - Unit I	Use R29-0167-0 Use R29-0242-1
X20-1729-3	IBM S/360 Model 20 Report Program Generator - Reference Card	
X20-1746-4	IBM S/360 Job Control Language Reference Card	Scrap X20-1729-2
X20-1769-0	OS/360 COBOL (F) & (F) Language Reference Card	Scrap X20-1746-3
X26-3566-4	IBM 1130 Reference Card	NEW
Y20-0167-0	TNL for 1130 Route Accounting System for Dairies and Bakeries (1130-DX-01X) System Manual Re: Y20-0078-0 with Y20-0121, Y20-0135 & Y20-0152	Scrap X26-3566-3 NEW
Y28-6618-1	IBM S/360 Operating System Indexed Sequential Access Methods PLM	
Z28-6649-2	IBM S/360: CLEAR Management Reporting System	Use Y28-6618-0
Z32-0351-1	Personal Training Guide - CST Student Guide	Scrap Z28-6649-1
Z32-0352-1	Personal Training Guide - BDP Student Guide	Scrap Z32-0351-0
Z77-8012-0	A Linear-programming technique to optimize traffic lane balance in a motor freight system	Scrap Z32-0352-0
Z77-8014-0	Minimum DOS documentation standards	NEW
Z77-8020-0	A hybrid computing system using the IBM 1800 data acquisition and control system and an analog computer	NEW NEW

PRL #12 March 22, 1968

H20-0507-0	Bibliography of Application Publications - Finance Industries	NEW
N20-1001.39	SRL NL Teleprocessing Re: A24-3089-6	Scrap N20-1001.38
N20-1077	TNL to Bibliography of Application Publications Finance Industries Re: H20-0507-0	NEW

MARKETING PUBLICATIONS

221-0485-0	IBM System/360 Administrative Terminal System Proposal Insert	NEW
520-0903-7	IBM System/360 Models 30, 40, 50, 65 and 75 Facts Folder	Scrap 520-0903-6
520-1554-0	For Cost-Conscious Management...A New Version of System/360 Model 20 Promotional Brochure	NEW
520-2075-0	Laboratory Automation Based Systems	NEW
542-0047-1	"Selectric" Instruction Book	Use 542-0047-0
543-0515-1	MT/ST Manual Mod IV	Use 543-0515-0
A22-6888-2	IBM System/360 Model 75 Configurator	Scrap A22-6888-1
A24-3232-3	IBM System/360 Mod 30 Configurator	Scrap A24-3232-2
A24-3332-4	IBM 29 Card Punch - Reference Manual	Use A24-3332-3
C26-3724-0	IBM 1800 Multiprogramming Executive Operating System Subroutine Library	NEW
C28-6586-7	IBM System/360 Operating System: Utilities	Use C28-6586-6 & N28-2298
E20-0286-0	IBM 1450 Bank Data Processing System for Demand Deposit Accounting	NEW
E20-0292-0	IBM 1450 Bank Data Processing System for Savings Accounting	NEW
E20-0293-0	IBM 1450 Bank Data Processing System for Mortgage Accounting	NEW
K20-0082-0	General Ledger Accounting Worcester County National Bank	NEW
K20-0251-0	Production and Job Cost at Madden Duplicating Services, Inc.	NEW
N22-0301	Changes to Data Acquisition Special Features for the IBM System/360 Model 44 Re: A22-6900-2	NEW
N26-0210	IBM 2841 Re: A26-5988-5	NEW
N30-5027	IBM System/360 Operating System: Basic Telecommunications Access Method Re: C30-2004-1	NEW
N33-1509	IBM System/360 Model 20 Functional Characteristics Re: A26-5847-2, -3 with N26-0174, N26-0176, N26-0179, N33-1501, N33-1504, N33-1506 & N33-1507	NEW
N33-1511	TNL to IBM System/360 Model 20 Installation Manual - Physical Planning Re: A26-5896-6 with N26-0164, N26-0180, N33-1500, N33-1502, N33-1503, N33-1508	NEW
Y20-0069-1	Attached Support Processor System (ASP) (360A-CX-15X) System Manual	Scrap Y20-0069-0 & Y20-0120
Z20-0016-0	Office Objectives	NEW
Z77-8019-0	Time Sharing Industry Survey	NEW
Z77-8021-0	1403 Computer Letter Writing on the Document Converter	NEW
Z77-8022-0	Measurement & Guidlines for Programmers a Method of Programming	NEW
Z77-8023-0	Conversion of FORTRAN Programs to S/360 FORTRAN	NEW
Z77-8025-0	Computer References by Category Prepared for DP Education	NEW
Z77-8026-0	Estimating DOS/COBOL Index Sequential File Times	NEW

April 19, 1968 Issue No. 68-07

IBM INSTALLATION NEWSLETTER

TABLE OF CONTENTS

OS RELEASE 14 PCP JOB SCHEDULER TERMINATING JOBS

The following is a field contribution which has not been submitted to any formal IBM test. Potential users should evaluate its usefulness in their own environment prior to implementation.

Effective with OS/Release 14, a "system ABEND " may occur at job termination when the following set of conditions exist:

1. A 64K machine is being used.
2. The 44K Job Scheduler has been specified.
3. An accounting routine has been specified and included (2600 + bytes).

As a result of the above, and changes made in the organization of the job termination modules, insufficient core is available to terminate jobs. In the case of one user, the PCP nucleus was 18K.

A solution is to re-organize and re-linkedit two job termination modules as shown in Exhibit 1 (for a S/360 Model 40).

OS RELEASE 14 PCP JOB SCHEDULER TERMINATING JOBS

```
                IEFZGJ01,                                                C
                IEFYSSMB,                                                C
                IEFZHEAK,                                                C
                IEFW31SD,                                                C
                IEFSD003,                                                C
                IEFDMSSS,                                                C
                IEFD3EAK,                                                C
                IEFSD007,                                                C
                IEFSD006,                                                C
                IEFZGMSG,                                                C
                IEFZHMSG,                                                C
                IEFACTLK,                                                C
                IEFACTRT,                                                C
                IEFWAD)
  ALIAS         IEFW23SD,IEF7A
  ENTRY         IEF7A
  NAME          IEFJTERM(R)
/*

//
//STAGETWO  JOB  83000,'PCP FOR MOD.40',MSGLEVEL=1
//SG2E EXEC PGM=IEWL,COND=(8,LT),                                  XXXXXXXX
//              PARM='NCAL,LIST,XREF,LET'
//SYSUT1 DD DISP=OLD,VOLUME=(,RETAIN),DSNAME=SYS1.UT3
//SYSPRINT DD SYSOUT=A
//SYSLMOD DD DISP=OLD,UNIT=2311,                                   XXXXXXXX
//              VOLUME=SER=111111,DSNAME=SYS1.LINKLIB
//MODLIB DD DISP=OLD,DSNAME=SYS1.MODLIB,VOLUME=(,RETAIN)
//SYSPUNCH DD DISP=OLD,VOLUME=(,RETAIN),DCB=(,RECFM=F,BLKSIZE=80),  XXXX
//              DSNAME=SYS1.OBJMOD
//SYSLIN DD *
  INCLUDE       MODLIB(IEFVDA,                                           C
                IEFVEA,                                                  C
                IEFVFA,                                                  C
                IEFVGI,                                                  C
                IEFVGK,                                                  C
                IEFVGMSS,                                                C
                IEFVGS,                                                  C
                IEFVGT,                                                  C
                IEFVHA,                                                  C
                IEFVHAA,                                                 C
                IEFVHB,                                                  C
                IEFVHC,                                                  C
                IEFVHCB,                                                 C
```

Exhibit 1

QS RELEASE 14 PCP JOB SCHEDULER TERMINATING JOBS (Continued)

```
                IEFVHE,                                              C
                IEFVHEB,                                             C
                IEFVHEC,                                             C
                IEFVHF,                                              C
                IEFVHGSS,                                            C
                IEFVHH,                                              C
                IEFVHL,                                              C
                IEFHMFAK,                                            C
                IEFVHQ,                                              C
                IEFVHRSS,                                            C
                IEFVJA,                                              C
                IEF23FAK,                                            C
                IEFQMSSS,                                            C
                IEF7KGXX,                                            C
                IEFSD008,                                            C
                IEFSD001,                                            C
                IEFSD007,                                            C
                IEEMCREK,                                            C
                IEFVEB,                                              C
                IEFSD006,                                            C
                IEF09FAK,                                            C
                IEFVDDUM,                                            C
                IEFSD090,                                            C
                IEFSD012)
    ALIAS       IEFVHA,IEFSD008,IEFKG,IEFVHAA
    ENTRY       IEFVHA
    NAME        IEFCNTRL(R)
    INCLUDE     MODLIB(IEF35DUM,                                     C
                IEFSD010,                                            C
                IEFW23SD,                                            C
                IEFZAJ83,                                            C
                IEFWTERM,                                            C
```

Exhibit 1

OS/360 AL ROUTINE TO CONVERT JULIAN TO GREGORIAN DATE

The following contribution has not been submitted to any formal IBM test. Potential users should evaluate its usefulness prior to implementation. The author has tested the routine only in an MVT environment.

Exhibit 1 contains an assembly listing of a subroutine that converts the date, as found on DS labels, and as returned by the TIME macro, to a human-oriented format.

The subroutine as written can be used directly only by ALP-(Assembly Language) coded callers. To be used by other source languages, it must be modified. To test the subroutine, the author coded an intermediary ALP program which receives a standard OS call, sets up the argument in GPRL, calls GREG, and returns the answer as required by the caller.

OS/360 FORTRAN ELIMINATE PROTECT RING RO TAPES

Newsletter 67-26 contained, on page 3, an article on the above subject. The Verify and REP entries were not properly sequenced. They should be "Verify" followed by REP, Verify fololwed by REP. SUPERZAP sets the "NO GO" switch only between "Verifies", which means that the proper sequence must be used whenever SUPERZAP is used for altering OS/360.

OS/360 AL ROUTINE TO CONVERT JULIAN TO GREGORIAN DATE

PAGE 1

```
STMT   SOURCE STATEMENT                                          F01JAN68   3/15/68

   2 GREG     START                                                        GREG0010
   3 *                                                                     GREG0020
   4 *                                                                     GREG0030
   5 * GREG - A SUBROUTINE TO CONVERT JULIAN (YYDDD) DATE TO GREGORIAN     GREG0040
   6 *        CODE IS REENTRANT                                            GREG0050
   7 *        RESTRICTION - INVALID FOR YEARS ENDING IN 00, EXCEPT AD 2000 GREG0060
   8 *        AT ENTRY ARGUMENT IS IN GPR1 IN FORM GIVEN BY TIME MACRO -   GREG0070
   9 *             00YYDDDS PACKED DECIMAL                                 GREG0080
  10 *        AT EXIT RESULT IS IN GPR1 -                                  GREG0090
  11 *             0YYMMDDS PACKED DECIMAL                                 GREG0100
```

```
STMT   SOURCE STATEMENT                                          F01JAN68   3/15/68

  13           USING *,15  CALLER HAS PRIMED BASE REGISTER                GREG0120
  14 DBASE     EQU   13  WILL WORK IN SAVE AREA                           GREG0130
  15 INDEX     EQU   4     FOR                                            GREG0140
  16 INCRMNT   EQU   2         BXLE                                       GREG0150
  17 LIMIT     EQU   3              USE                                   GREG0160
  18 DAYJR     EQU   1  JULIAN DATE INHERITED FROM CALLER                 GREG0170
  19           USING DATA,DBASE    DATA POINTER                           GREG0180

  21           SAVE  (14,5)                                               GREG0200
  22+          DS    0H
  23+          STM   14,5,12(13) SAVE REGISTERS
  24           MVN   DAYJ+L'DAYJ-1(1),TBL+L'TBL-1  MATCH UP ZONES         GREG0210
  25           TM    YRJ,X'13' BEGIN TO CHECK FOR LEAP YEAR               GREG0220
  26           BZ    LEAP  00, 04, 08, ¤MOD 20¤                           GREG0230
  27           BO    COMMON  13, 17 ¤MOD 20¤                              GREG0240
  28           TM    YRJ,X'12'                                            GREG0250
  29           BNO   COMMON  ALL OTHERS EXCEPT 12, 16 (MOD20)             GREG0260
  30 LEAP      CLC   DAYJ,=P'60'    FEB 29                                GREG0270
  31           BL    COMMON  EARLIER - TREAT AS IN A COMMON YEAR          GREG0280
  32           BE    FEB29  SPECIAL CASE                                  GREG0290
  33           SP    DAYJ,TBL+2*L'TBL-1(1)  DAY=DAY-1 TO SIMULATE COMMON YEAR GREG0300
  34 COMMON    MVO   YRG,YRJ  MAKE ROOM                                   GREG0310
  35           NI    YRJ,X'F0'          FOR MONTH                         GREG0320
  36           LM    INCRMNT,INDEX,=A(L'TBL,11*L'TBL,L'TBL)               GREG0330
  37           LH    DAYJR,DAYJ                                           GREG0340
  38 LPA       CH    DAYJR,TBL(INDEX)     TABLE                           GREG0350
  39           BNH   KALEND                    SEARCH                     GREG0360
  40           BXLE  INDEX,INCRMNT,LPA                                    GREG0370
  41 KALEND    LA    LIMIT,TBL-L'TBL(INDEX) POINT TO SUBTRAHEND IN TABLE  GREG0380
  42           SP    DAYJ,0(L'TBL,LIMIT) SUBTRACT CUMULATIVE DAYS         GREG0390
  43           SRDL  INDEX,33  PREPARE TO DIVIDE                          GREG0400
  44           D     INDEX,=F'10'  CONVERT MONTH TO PACKED DECIMAL        GREG0410
  45           EX    INDEX+1,OIT  TENS DIGIT OF MONTH                     GREG0420
  46           SLL   INDEX,4  POSITION UNITS DIGIT                        GREG0430
  47           EX    INDEX,OIU  UNITS DIGIT OF MONTH                      GREG0440
  48           OI    DAYG+1,X'0F'         PLUG ZONE                       GREG0450
```

Exhibit 1

OS/360 AL ROUTINE TO CONVERT JULIAN TO GREGORIAN DATE (Continued)

```
49 BACK      RETURN  (14,5),T                                          GREG0460
50+BACK      DS      0H
51+          LM      14,5,12(13) RESTORE THE REGISTERS
52+          MVI     12(13),X'FF' SET RETURN INDICATION
53+          BR      14 RETURN
54 FEB29     EX      0,COMMON   MAKE ROOM                              GREG0470
55           NI      YRJ,X'F0'         FOR MONTH                       GREG0480
56           MVC     MTH(2),=X'229F' DATE IS FEB 29                    GREG0490
57           B       BACK                                              GREG0500
58 OIT       OI      MTH,*-*  PLUGS TENS DIGIT OF MONTH                GREG0510
59 OIU       OI      MTH+1,*-*  PLUGS UNITS DIGIT OF MONTH             GREG0520

STMT   SOURCE STATEMENT                                    F01JAN68   3/15/68

 61 *                  CONSTANTS AND DATA DEFINITIONS                  GREG0540

 63 TBL       DC       PL2'0,31,59,90,120,151,181,212,243,273,304,334' GREG0560

 65 DATA      DSECT                                                    GREG0580
 66           DS       6F   BEGINNING OF SAVE AREA                     GREG0590
 67 DATE      DS       0F                                              GREG0600
 68 DATEG     DS       0PL4                                            GREG0610
 69 YRG       DS       0PL2                                            GREG0620
 70           DS       PL1                                             GREG0630
 71 DATEJ     DS       0PL3                                            GREG0640
 72 YRJ       DS       0PL1                                            GREG0650
 73 MTH       DS       0PL2                                            GREG0660
 74           DS       PL1                                             GREG0670
 75 DAYJ      DS       0PL2                                            GREG0680
 76 DAYG      DS       PL2                                             GREG0690
 77           END                                                      GREG0700
 78                    #A(L'TBL,11*L'TBL,L'TBL)
 79                    #F'10'
 80                    #P'60'
 81                    #X'229F'
```

Exhibit 1

DOS/360- OS DUMP-RESTORE V- 6 INCORRECT DOS PACK BACKUP TAPES

While some installations use both OS/360 and DOS/360, other DOS/360 users should not be encouraged to order and use this OS/360 utility if it is not required in their installation. The following article is intended for those users who presently use the OS/360 utility for generating backup tapes for DOS/360 packs.

When a DOS/360 pack is dumped to tape by use of the Version 6 OS/360 Dump-Restore deck, the standard label cylinder will not always be handled correctly. The trouble becomes apparent only when the pack is later restored from the tape, and a message is printed that the program is terminated because of an I/O check. What happens is that everything up to the standard label cylinder is put back on the pack.

Version 5 of OS/360 Dump-Restore will operate correctly. This intermittent problem appears isolated to using Version 6 in going from a DOS/360 pack to a tape. Neither Version, 5 nor 6, will bring a tape back to a pack correctly if the tape was written by Version 6. Both versions will bring a tape back to a pack correctly if the tape was written by Version 5.

The following procedure is believed to be a solution: If you have created DOS backup tapes under Version 6, proceed as usual to restore the pack, then when the job stops (at the standard label cylinder), re-IPL from that pack and submit a job with //OPTION STDLABEL to recreate the standard cylinder. You must also reinitialize the DSR cylinder, then create your backup tapes with Version 5 Dump-Restore. (Make sure the pack has been ititialized to meet DOS/360 needs.)

DOS/360 RPG PROCESSING UNLABELED MULTI-REEL INPUT FILES

The following comment concerns an article with the above title which appeared in Newsletter issue 67-25 (12/15/67), page 26. This is a field contribution and has not been submitted to any formal IBM test. Potential users should evaluate its usefulness in their own environment prior to implementation.

Restriction number 2 ('Matching Records must not be a requirement for processing the files') is not really a restriction. Since it is possible to match records on more than two files, the multi-reel file can be matched as two separate files against a third file required for processing.

A user had an RPG program with two unlabeled input files and matching records were used in processing them. When one of the files went to two reels, he called each reel a separate file (both secondary) and the other file the primary file, matching records on all three.

DOS/360 SORT BLOCK AND RECORD COUNTS

GPR 1 points to the DTF containing the block counts. Refer to the DOS/360 LIOCS manual for the DTF format.

At Exit 32, at output time, GPR 10 contains the record count.

S/360-20 RPG SELECTIVE CHAINING

The following is a field contribution which has not been submitted to any formal IBM test. Potential users should evaluate its usefulness in their own environment prior to implementation.

Selective Chaining on the Model 20 may be accomplished by an indicator in the Field Record Relation columns of the Input Specifications. This indicator is different from the one specified as Resulting Indicator (Col. 19 and 20).

A typical example is billing, where the input records contain customer number, part number and quantity. The chaining required is to the customer name and address file and the master inventory file. Without added user coding, RPG will chain to both customer name and address file, and the master inventory file for every detail record read. However, chaining to the customer name and address file is only required on the first card of a group. The example in Exhibits 1 through 4 shows how this is accomplished.

The author states that one Model 20 disk user, increased his billing speed by 50% using this method. This is not to be construed that all users would obtain the same improvement.

REPORT PROGRAM GENERATOR OUTPUT-FORMAT SPECIFICATIONS

IBM System 360

Line	Form Type	Filename	Type (H/D/T)	Stacker Select	Space Before	Space After	Skip Before	Skip After	Output Indicators And	And	And	Field Name	Zero Suppress (Z)	Blank After (B)	End Position in Output Record	Packed Field (P)	Constant or Edit Word
									Not	Not	Not						
3 4 5	6	7 8 9 10 11 12 13 14	15	16	17	18	19 20	21 22	23 24 25	26 27 28	29 30 31	32 33 34 35 36 37	38	39	40 41 42 43	44	45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70
0 1	O	INVOICE	H						Ø1	Ø3							
0 2	O	* PRINT		I	N	V	O	I	C	E	HEADING						
0 3	O		D						Ø1	Ø4							
0 4	O	* PRINT		I	N	V	O	I	C	E	LINE ITEM (BODY).						
0 5	O		T						L1	6Ø							
0 6	O	* PRINT		I	N	V	O	I	C	E	TOTAL.						
0 7	O																

IBM System/360 Report Program Generator Indicator Summary

Circle Indicators Used:

01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20
21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60
61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80
81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	OF
L0	L1	L2	L3	L4	L5	L6	L7	L8	L9	LR	M1	M2	M3	H1	H2	C1	C2	C3	OV

Line	Form Type	Indicators: Input Resulting	Calc. Resulting	Field	Matching & Chaining	Ctl. Break & Halt	FUNCTION OF INDICATORS
3 4 5	6 7	8 9 10	11 12 13	14 15 16	17 18 19	20 21 22	23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72
0 1	F *	I	C	F	M	L	FUNCTION OF INDICATORS
0 2	F *		99		C1		CONTROLS CHAINING TO CUSTOMER MASTER FILE.
0 3	F *				C2		CHAINS TO INVENTORY MASTER FILE.
0 4	F *	Ø1					ITEM CARD.
0 5	F *	Ø2					CONTROL CARD (DATE AND INVOICE NUMBER)
0 6	F *	Ø3					CUSTOMER MASTER RECORD
0 7	F *	Ø4					INVENTORY MASTER RECORD
0 8	F *		5Ø				CONTROLS FALSE L1 CAUSED BY CONTROL CARD.
0 9	F *		6Ø				CONTROLS FALSE L1 CAUSED BY CONTROL CARD.
1 0	F *					L1	INDICATES CUSTOMER CHANGE - TOTAL INVOICE
1 1	F *						

Exhibit 1

IBM

INTERNATIONAL BUSINESS MACHINES CORPORATION

REPORT PROGRAM GENERATOR INPUT SPECIFICATIONS

IBM System/360

Date _____

Program SELECTIVE CHAINING

Programmer _____

Punching Instruction	Graphic								
	Punch								

Page 04

Program Identification CHAIN

Line	Form Type	Filename	Sequence	Number (1-N)	Option (O)	Resulting Indicator	Record Identification Codes 1: Position	1: Not (N)	1: C/Z/D	1: Character	2: Position	2: Not (N)	2: C/Z/D	2: Character	3: Position	3: Not (N)	3: C/Z/D	3: Character	Stacker Select	Packed (P)	Field Location: From	Field Location: To	Decimal Positions	Field Name	Control Level (L1-L9)	Matching Fields or Chaining Fields	Field-Record Relation	Field Indicators: Plus	Field Indicators: Minus	Field Indicators: Zero or Blank	Sterling Sign Position
01	I	DETCD	AA			01	1	N	C	D																					
02	I																				1	50		CUSTNO	L1						
03	I																				1	50		CHAIN		C1	99				
04	I																				6	10		PARTNO		C2					
05	I																				11	14		QTY							
06	I		BB			02	1		C	D																					
07	I																				2	7		DATE							
08	I																				8	12		INVNO							
09	I	* THE CHAINING FIELD (CUSTNO) MUST BE REDEFINED. WHEN L1, C1, AND																													
10	I	* 99 ARE SPECIFIED ON THE SAME LINE AN ERROR OCCURS.																													
11	I	CUSTOMER	AA			03																									
12	I	* CUSTOMER NAME AND ADDRESS FIELDS			N																										
13	I	INVENT	AA			04																									
14	I	* INVENTORY MASTER RECORD.																													
15	I																														
	I																														
	I																														
	I																														
	I																														
	I																														

S/360-20 RPG SELECTIVE CHAINING

Exhibit 2

INTERNATIONAL BUSINESS MACHINES CORPORATION

REPORT PROGRAM GENERATOR FILE DESCRIPTION SPECIFICATIONS

IBM System/360

Line	Form Type	Filename	File Type: I/O/U/C	P/S/C/R/T	E	File Designation: A/D	End of File: F/V	Sequence	File Format: Block Length	Record Length	Mode of Processing: L/R	Length of Key Field or of Record Address Field	Record Address Type: K/I	Type of File Organization: I/D/T	Overflow Indicator	Key Field Starting Location	Extension Code E/L	Device	Symbolic Device	Labels (S, N, or E)	Name of Label Exit	Extent Exit for DAM	File Addition: A	No. Tracks for Cylinder Overflows	No. of Extents	Tape Rewind: N/U
01	F	DETCD	I	P		A	F		80	80							E	READ01								
02	F	CUSTOMER	I	C			F		270	270			K	I		2		DISK11F	SYS001	S					3	
03	F	INVENT	I	C			F		270	90			K	I		2		DISK11F	SYS002	S					3	
04	F	INVOICE	O				F		132	132								PRINTER								
05	F																									

Form X24-3348-1 U/M025
Printed in U.S.A.

REPORT PROGRAM GENERATOR FILE EXTENSION SPECIFICATIONS

IBM System/360

Line	Form Type	Record Sequence of the Chaining File: Number of the Chaining Field	From Filename	To Filename	Table Name	Number of Table Entries Per Record: Number of Table Entries Per Table	Length of Table Entry: Packed (P)	Decimal Pos.	Sequence (A/D)	Table Name (Alternating Table)	Length of Table Entry: Packed(P)	Decimal Pos.	Sequence (A/D)	Comments
01	E	C1	DETCD	CUSTOMER										
02	E	C2	DETCD	INVENT										
03	E													
04	E													
05	E													
06	E													
07	E													
08	E													
09	E													
10	E													

TITLE

Exhibit 3

IBM

INTERNATIONAL BUSINESS MACHINES CORPORATION

REPORT PROGRAM GENERATOR CALCULATION SPECIFICATIONS

IBM System/360

Date _____

Program SELECTIVE CHAINING

Programmer _____

Punching Instruction	Graphic							
	Punch							

Page 05

Program Identification CHAIN

Line	Form Type	Control Level (L0-L9, LR)	Indicators: And / Not	And / Not	And / Not	Factor 1	Operation	Factor 2	Result Field	Field Length	Decimal Positions	Half Adjust (H)	Resulting Indicators: Plus	Minus	Zero or Blank	Compare: High 1 > 2	Low 1 < 2	Equal 1 = 2	Comments
3 4 5	6	7 8	9 10 11	12 13 14	15 16 17	18 19 20 21 22 23 24 25 26 27	28 29 30 31 32	33 34 35 36 37 38 39 40 41 42	43 44 45 46 47 48	49 50 51	52	53	54 55	56 57	58 59				60 61 62 63 64 65 66 67 68 69 70 71 72 73 74
01	C						SETOF						99						
02	C	*	STOPS	CHAINING	TO	CUSTOMER MASTER	FILE.												
03	C																		
04	C																		
05	C																		
06	C	*	DETAIL	CALCULATIONS															
07	C																		
08	C																		
09	C																		
10	C	L1					SETON						99						
11	C	*	ALLOWS	CHAINING	TO	CUSTOMER FILE	WITH	FIRST	RECORD										of NEXT GROUP.
12	C	L1	50				SETON						60						
13	C	L1					SETON						50						
14	C	*	SUPPRESSES	FALSE	L1	CAUSED	BY	DATE	CARD.										
15	C	*	ALL	L1	OUTPUT	CONTROLLED	BY	L1	AND	60.									
	C																		
	C																		
	C																		
	C																		
	C																		

Card Electro Number _____

Exhibit 4

NEW TYPE III PROGRAMS *

The following are the abstracts of programs which have been recently made available from the Type III library.

The program, along with its complete abstract, will be incorporated in subsequent issues of the Catalog of Programs.

Programs may be obtained by submitting a properly completed "General Program Request Card" (Form Number 120-1145-1) to the Program Information Department, 40 Saw Mill River Road, Hawthorne, New York, 10532.

These programs and their related documentation are distributed by IBM in the author's original form and have not been subjected to any formal testing.

Any discussions of Type III programs with customers must emphasize the following points

1. Type III (IBM employee--contributed programs) are provided by the IBM company as part of its service to customers. They are not a part of the IBM product line support.

2. Type III programs have not been subjected to any formal test.

3. Recipients of Type III programs are expected to make the final evaluation as to the usefulness of the programs in their own environment.

4. There is no committed maintenance for Type III programs. However, any changes the author chooses to make will be announced in subsequent issues of the Catalogs of Programs.

5. IBM makes no warranty, expressed or implied, as to the documentation, function, or performance of Type III programs.

*NOTE: THE CUSTOMER MUST BE INFORMED THAT THE ABOVE APPLIES TO ANY OF THE FOLLOWING TYPE III ABSTRACTS FURNISHED TO HIM.

1401 GENERAL PROCESSING UTILITY PROGRAM

This program abstract was published in Newsletter issue 68-05, page 20. The program order file number was omitted. It is 1401-01.4.232. Please enter it in your copy of the Newsletter if you will require reference to it in the future.

S/360 SEQUENTIAL DATA SET LISTER PROGRAM. An OS/360 Assembler Language Program whose function is to list logical records from a Sequential Data Set. Logical records are listed 100 characters per line group for as many line groups as are configured as follows:

Line 1	Character Representation (If Printable)
Line 2	HEX Zone
Line 3	HEX Digit
Line 4	Delineator Line (Numbered Every 5 Spaces)

The line group format is convenient for examining unprintable data and for locating particular bytes within a record by means of the Delineator Line. The program reads records with QSAM and produces the listing with QSAM. Input sequential record formats supported are: Fixed, Fixed Blocked, Variable, Variable Blocked.

Ordering Procedure: Order File Number 360D-04.1.003.

The program material can be obtained on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

If not specified, a 9 track DTR will be provided.

No tape submittal is required. The DTR will be provided by the Library.

S/360 INDEX SEQUENTIAL DATA SET LISTER PROGRAM. An OS/360 Assembler Language Program whose function is to list logical records from an ISAM Data Set. Logical records are listed 100 characters per line group for as many line groups as are required to display an entire logical record. The line group format is configured as follows:

Line 1	Character Representation (If Printable)
Line 2	HEX Zone
Line 3	HEX Digit
Line 4	Delineator Line (Numbered Every 5 Spaces)

The line group format is convenient for examining unprintable data and for locating particular bytes within a record by means of the delineator line. The program reads records with QISAM scan mode and produces the listing with QSAM. Input ISAM record formats supported are: Fixed-Fixed Blocked.

Ordering Procedure: Order File Number 360D-04.1.004.

The program material can be obtained on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

If not specified, a 9 track DTR will be provided.

No tape submittal is required. The DTR will be provided by the Library.

TEST DATA SET GENERATOR PROGRAM - OS/360. This program converts input control cards and data to sequential data set(s). Output record formats can be F, FB, V, VB. Input to the DSG can be cards, card images on tape, DASD or PDS members. Translated output DS(s) can reside on tape, DASD, cards or any combination thereof. A maximum of 99 Data Sets can be generated in a single execution. Maximum length of each output logical record is 4095 bytes. DSG produces binary, packed decimal, EBCDIC, Hexadecimal, fill, Binary Incrementing/Decrementing, packed decimal Incremeting/ Decrementing fields in any combination or order. All output records are automatically listed in a numbered character/hex 4-line format as they are being generated. A 'NOLIST' option allows supression of the data set print out. DSG has the capability of repeating entire records any number of times or repeating selected fields for any duration. Error checking and diagnostic routines fully document all errors to include type of error, card number in error, field number in error and appropriate corrective action. DSG is particularly easy to use as there is only one type of control card and input data is coded in free format positional independent notation. Language=OS/360 BAL-Config=any OS/360 system w/dec. feat.

Ordering Procedure: Order File Number 360D-04.3.004.

The program material can be obtained on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

If not specified, a 9 track DTR will be provided.

No tape submittal is required. The DTR will be provided by the Library.

S/360 THREE D'S - DUMP, DISASSEMBLE, AND DELINKEDIT. The three D's are a series of related programs which allow the user to convert machine language to basic assembly language. The disassembler accepts object decks composed of single CSECTS and translates them to basic assembly language decks or listings. DISINF is a subroutine which enables the user to interface with the disassembler to produce mnemonic dumps of core. The delinkeditor allows him to convert load modules to object modules and then to source decks through the disassembler. Routines DISINF,

the disassembler, and delinked require 1K, 64K, and 3K of core respectively. All three routines are designed to execute under OS/360, though with modifications they could execute under DOS. All three programs are written in Basic Assembly Language. Minimum memory size - 64K; card reader, punch, printer.

Ordering Procedure: Order File Number 360D-04.1.005.

The program material can be obtained on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

If not specified, a 9 track DTR will be provided.

No tape submittal is required. The DTR will be provided by the Library.

S/360 FORTRAN LINEAR PROGRAMMING SYSTEM. This program solves linear programming problems using "contracted tableau" form of the simplex method. Input to the program is in standard Share format (Title Card, ROWID section, MATRIX section, and FIRST B section). The problem is solved in core, and the output includes an iteration log, optimal solution and cost range report, and reduced costs of slack variables and structural variables not in the optimal solution. The program, as distributed, can solve a problem of size MxN up to 40x40 (where M=number of rows, N=number of structural columns plus number of $>$ rows), on a 32K or larger System/360 under BPS, DOS, TOS or OS FORTRAN. Sample running time for problems of size 25x25 is of the order of 1 minute on a Model 30.

Ordering Procedure: Order File Number 360D-15.2.006.

The program material can be obtained in card form or on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR). If not specified, a 9 track DTR will be provided.

No tape submittal is required. The DTR will be provided by the Library.

S/360 ALP DOS DECISION TABLE MACROS. A set of two macros for the coding and execution of limited entry decision tables in assembly language under DOS or OS. one macro, "SETIT", , sets switches which correspond to the condition entries of the decision tables. The other macro, "TABLE", generates full word constants that are analogs of the decision table rules, and provides linkage to user written sub-routines that correspond to the action statements. It also generates and provides linkage to, an off-line sub-routine. This subroutine searches a descision table for the first rule whose conditions are satisfied, and then executes the action statement sub-routines indicated by that rule. These macros make available the logical benefits of decision tables at the ALP coding level. They also augment documentation clarity, facilitate modular programming, and aid de-bugging, Machine configuration: 16K 360 model 30. Programming systems: DOS. source language: ALP.

Ordering Procedure: Order File Number 360D-03.7.017.

The program material can be obtained in card form or on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR). If not specified, a 9 track DTR will be provided.

No tape submittal is required. The DTR will be provided by the Library.

S/360 DUCS - IBM 2260 DISPLAY UNIT CONTROL SYSTEM. This system will provide complete support for local IBM 2260 low cost display units, including the data entry models under control of either the Tape Operating System or the Disk Operating System (Vers. II or VERS. III). It can be used by problem programs written in PL/I, COBOL, or Assembler Language and is referenced by standard CALL statements. The user need not have any knowledge of Assembler Language to implement DUCS. With DUCS, the need for BTAM for the support of locally attached 2260 display units is eliminated. Also teleprocessing support need not be specified in the user tailored supervisor . Thus, allowing the user to specify an 8K system to support 2260 display units in a multiprogramming environment. Any number of display units can be run with DUCS.

Also, asynchronous interrupt handling along with many other features not found in BTAM are intrinsic functions of DUCS.

Ordering Procedure: Order File Number 360D-00.6.006.

The program material can be obtained on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR). If not specified, a 9 track DTR will be provided.

No tape submittal is required. The DTR will be provided by the Library.

S/360 FACILITY SUBROUTINES. The S/360 Facility Subroutines were written to give the S/360 programmer greater magnitude in the use of S/360 features not supported by the specific language translators being used. The subroutines may be used by any of the language translator object programs under the Tape and Disk Operating Systems. The subroutines utilize the standard Assembler Language macros to permit the higher level language programs to have access to the storage print facility, the communication region dates, the interval timer, the User Program Switch Indicator (UPSI byte) and the external interupt key.

Minimum Machine Configuration: With the exception of the interval timer subroutines and the subroutine to ring the audible alarm, all of which require the interval timer feature, there is no minimum configuration requirement.

Ordering Procedure: Order File Number 360D-01.0.004.

The program material can be obtained in card form or one one 9 track or 7 track (Data Conversion feature requied) Distribution Tape Reel (DTR).

If not specified, a 9 track DTR will be provided.

No tape submittal is required. The DTR will be provided by the Library.

CALCDASD - A S/360 TOS/DOS COBOL PROGRAM TO CALCULATE NUMERICAL CHARACTERISTICS OF FILES FOR DASD UNITS FOR GIVEN RECORDS. This program will compute the number of characters, logical records and blocks per track, for a blocking factor of 1 to the maximum permissible, for any of the supported devices. The DASD supported are 2301, 2302, 2303, 2311, 2314 and 2321. The program as distributed requires a minimum of 64K; however, alterations can be easily made to allow operation on a 24K or 32K system.

Ordering Procedure: Order File Number 360D-00.5.002.

The program material can be obtained in card form or on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

If not specified, a 9 track DTR will be provided.

No tape submittal is required. The DTR will be provided by the Library.

S/360 RENUMBER - A SELF-RELOCATING UTILITY PROGRAM TO REPRODUCE AND RESEQUENCE ALL DOS/TOS ASSEMBLER LANGUAGE, COBOL, PL/I, RPG AND FORTRAN PROGRAMS. RENUMBER is a self-relocating DOS/TOS utility program which will resequence and reproduce source program decks written in the following DOS/TOS languages: Assembler Language; COBOL; PL/I; RPG, and FORTRAN. Also, Assembler Language programs will have the Name, Operation and Operand fields realigned to columns 1, 10, and 16 where necessary along with continuation indicators in successive cards being sequentially numbered. Futhermore, ISEQ 73, 80 instructions are inserted in the beginning of each Assembler deck and all previous ISEQ instructions are ignored. It is not necessary for the user to prepare any special control cards for input to RENUMBER nor is it necessary to remove any Job Control statements associated with the source program deck. The identification information for any of the five types of source programs is taken from the respective positions in the // EXEC translator statement.

Ordering Procedure: Order File Number 360D-00.0.010.

The program material can be obtained in card form or on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

If not specified, a 9 track DTR will be provided.

No tape submittal is required. The DTR will be provided by the Library.

SYSTEM/360 MODEL 20 CARD LOW-LEVEL CODE GENERATOR SYSTEM. This System is used to generate low-level coding which is used in material planning applications. Low level coding if used and maintained properly, facilitates netting gross requirements against inventory balances. Eliminating the premature allocation of stock when used in conjunction with lead time. Programs are written in RPG and require the following machine specifications: 8K 2020, 2501, and 2203.

Ordering Procedure: Order File Number 360E-23.3.004.

The program material can be obtained in card form or on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

If not specified, a 9 track DTR will be provided.

No tape submittal is required. The DTR will be provided by the Library.

TRANSPORTATION PROGRAM FOR THE IBM 1130. This program solves the transportation problem, i.e., an optimal solution is obtained to a special class of linear programming problems where "surpluses exist at the originating sites (sources) and are to be transported to locations (destinations) with "shortages". The mathematical method of solution is that developed by Fort and Fulkerson (Management Science 3 (1): 24-32, October, 1956; and, Solving the Transportation Problem, The Rand Corporation, P-895, June 20, 1956).

The program is a FORTRAN Version of the IBM 1620/SPS/TAPE by Madden and Smith, File Number 1620-10.1.003. The program is written in FORTRAN to run under the 1130 monitor on an 8K, card, 1132 printer and disk system.

Ordering Procedure: Order File Number 1130-15.2.004.

Distribution will be in card form only.

IBM Installation Newsletter

May 17, 1968

Issue No. 68-09

CONTENTS

Distribution: Branch Offices - DP Management, Salesmen, Systems Engineers, FE Managers. Regions, Districts, Education Centers, Field Systems Centers, Federal Systems Centers, FE Area Offices, DPD HQ, FED HQ, WTC.

*Requires Immediate Attention

For IBM Internal Use Only

SELECTED SHORT SUBJECTS

The purpose of Selected Short Subjects is to bring together and highlight concise, factual and timely information which will indicate that action is to be taken by the IBM representative whose accounts are affected.

1. Teleprocessing - TELPAK Tariff Revisions

An article in Installation Newsletter issue 68-07, page 9, described revisions which were to take effect on June 1, 1968. The reaction of users to these significant changes led to a suspension of the effective date until September 1, 1968. The FCC has stated that this date is firm (only court action can further delay it). Please mark the article in your copy of 68-07 to show this change.

In a further development, the FCC has reaffirmed the temporary level of these rates, stating that later changes will probably be prescribed by the FCC and will become effective without delay.

2. 1287 OCR Planning and Installing Information

If you are interested in the 1287 OCR area, you should review the article with the above title in this issue of the Installation Newsletter.

3. Type III Programs in Development

Type III Programs in development were formerly contained in the KWIC Index to TIE. They are now contained in SECOM which provides an information broadcast and retrieval function for the Branch Offices. All new Type III programs are announced when they become available, and their abstracts are published, in the IBM Installation Newsletter. The abstract includes the necessary ordering information.

No orders should be sent to PID until the programs are announced as being available in the Newsletter.

OS/360 RELEASE 14 CHANGE NON-STANDARD CHARACTERS IN JCL CORRECTION

Newsletter issue 68-07 contained, on page 2, an article on the above subject. The following is a correction to that article.

The following article concerns changes to Release 14 of OS/360 to handle non-standard characters in the JCL DSNAME and VOLUME Serial number fields. These changes are temporary bypasses for those who have been using the nonstandard characters. The general problem of allowing the characters which were originally specified as invalid in the JCL manual, but not diagnosed by earlier versions of the Reader/Interpreter is being reviewed. The following modifications are to module IEFVFA:

1. To permit a +0 (12 and 0 punches) in the DSNAME field;

```
Verify 061C 4120, 0017
REP    061C 47F0, B5C4
```

2. To permit a - (hyphen or 11 punch) in the DSNAME field;

```
Verify 081A 5010 80B8
REP    081A 4700 0000
```

3. To permit a - (hyphen or 11 punch) in the VOLUME Serial number field;

```
Verify 0826 4770 B830 9500 6000
REP    0826 4780 B504 951E 5000
```

OS/360 - IDENTIFYING DDNAMES OF MISSING DDCARDS

The following is a field contribution which has not been submitted to any formal IBM test. Potential users should evaluate its usefulness in their own environment prior to implementation. The author states that the technique has proven to be useful in quickly identifying the OC1, etc. ABENDS caused by missing ddcards.

NOTE: Type I code is being added to the module IGC0001I for Release 15/16, so the user must revise this approach if using a release subsequent to Release 14. Also, changes in future releases of OS/360 may affect this technique or negate the need of it.

Exhibit 1, illustrates a SUPERZAP patch to the OPEN executor module IGC0001I that causes it to issue a message for each missing ddcard representing a dataset that the user program attempts to open. This explicitly identifies ddnames of missing ddcards, which otherwise usually results in ABEND with no explicit indication of the cause of the problem.

One instruction, a BC 8, is overlaid in module IGC0001I at location 7A (hex) causing the conditional branch to go to additional code placed in the module's free area. This branch is taken when a ddname-not-found condition exists after TIOT scan. The eighteen added instructions first issue GETMAIN for a message text area, then build a message of the form:

"IEC999I DDCARD MISSING, jobname stepname. procstep, ddname"

WTO is issued, core freed, and control returned to the original branch destination. The user could optionally change the message text, abnormally terminate the task, or exit the task by placing the task return address into the RB resume PSW field.

Exhibit 2 shows the message text on a console log sheet.

OS/360 - IDENTIFYING DDNAMES OF MISSING DDCARDS

```
//FIXXX JOB  MSGLEVEL=1,REGION=100K
//ZAP  EXEC  PGM=SUPERZAP
//SYSLIB DD DSNAME=SYS1.SVCLIB,DISP=OLD
//SYSIN DD *
 NAME IGC0001I
 DUMP IGC0001I
 VERIFY 7A 47EC3C1E     VERIFY MAIN ROOK INTO REPS
 REP    7A 47803318  SET BC 8 TO NEW CODE(END OF TIOT SCAN)
 REP C3CC 00FC0000            WTO PARM FIELD (300 IS START OF FREEAREA)
 REP C3C4 C9C5C3E9E9E9C9C4C4C4C4C3  MSGTXT='IEC999I  DDC'
 REP C31C   C1D9C44CD4C9E2E2C9D5C76B  MSGTXT='ARD MISSING'
 REP C31C 41000064            LA    0,100 LOAD GETMAIN COUNT
 REP C320 45103322            BAL   1,*+4 LOAD-R1 FOR SVC, INC. GETMAIN
 REP C324 0A0A                SVC   10  ISSUE GETMAIN SVC FOR MSGAREA
 REP C326 18D1                LR    13,1   SAVE GETMAIN AREA ADDR
 REP C328 D21BD000C32FE       MVC   0(28,13),766(3) MSGTXT TO MSGAREA
 REP C32E D2C7D01C9000        MVC   28(8,13),0(9) JOBNAME TO MSG
 REP C334 924BDC24            MVI   36(13),C'.' DOT AFTER JOBNAME
 REP C338 D2C7D025C008        MVC   37(8,13),8(9) STEPNAME TO MSG
 REP C33E 924BDC2D            MVI   45(13),C'.' DOT AFTER STEPNAME
 REP C342 D2C7D02E901C        MVC   46(8,13),16(9) STEPNAME TO MSG
 REP C348 926BDC36            MVI   54(13),C',' COMMA TO MSG
 REP C34C D2C7D037202E        MVC   55(8,13),40(2)  DDNAME TO MSG
 REP C352 0A23                SVC   35  WTO MESSAGE
 REP C354 41000064            LA    0,100   LOAD REG FOR FREEMAIN
 REP C358 181D                LR    1,13  SET FREE ADDR
 REP C35A 0A0A                SVC   10  FREEMAIN MSGAREA CORE
 REP C35C 17DD                XR    13,13  CLEAR R13 BEFORE RETURN
 REP C35E 47FC3C1E            BC    15,28(0,3)  RETURN TO MAIN CODE
 DUMP IGC0001I
/*
```

Exhibit 1

OS/360 - IDENTIFYING DDNAMES OF MISSING DDCARDS

```
IEF233A M 182,ETA1   ,ORBIT
IEC999I  DDCARD MISSING,ORBIT    .EXEC     .          ,SYSABS
IEC999I  DDCARD MISSING,ORBIT    .EXEC     .          ,SYSPUNCH
ESTIMATED LINES EXCEEDED
IEF280I K 180,       ,ORBIT
IEF280I K 181,       ,ORBIT
IEF280I K 182,       ,ORBIT
IEF450I ORBIT   .EXEC     .          ABEND S0C1
```

Exhibit 2. Two ddcards missing; job ABENDED.

OS/360 BTAM DFTRMLST MACRO BSC MULTIPOINT

When specifying multipoint polling lists under Binary Synchronous Communications, a last dummy entry must be EOT characters coded in the line transmission code, and equal in length to the other entries.

EXAMPLE: (EBCDIC)

X	- C1 - C1 - 2D	(A A Inq)
X	-C2 - C2 - 2D	(B B B Inq)
X	-37 - 37 - 37	(EOT EOT EOT)

OS/360 QTAM FREEING OF GET/PUT CORE AT CLOSE

The core occupied by GET/PUT Modules is not being immediately freed at CLOSE time due to a failure to DELETE them in the Process Close Module. Specific difficulties (if any) depend on the system environment. Presently, consideration is being given to correcting this problem in a future OS/360 Release. If this impacts the user's operation in the interim, he can DELETE these modules (one needed for each DCB being closed) following execution of his CLOSE to insure proper system control.

OS/360 FORTRAN EXTENDED ERROR MESSAGE FACILITY

The following information applies to OS/360 (Release 16) as announced in P68-59 and is presented here for planning purposes only. The actual availability and specifications will be as provided through the official IBM procedures, the Program Announcement Letter and formal documentation.

The FORTRAN Extended Error Message Facility is chosen at System Generation time by the OPTERR parameter in the FORTLIB macro for a G or H level library. The user will benefit by more informative messages, an improved Traceback, and continued execution with a standard fixup. The user can also dynamically control the number of erros to allow and number of messages to print, and he can suppress Traceback. He can fix up illegal data in a user written subroutine and call for a Traceback at any point. Traceback now will print registers 14-1 and start at the routine detecting the error.

The following is a list of points to consider when implementing:

1. As stated in the Programmer's Guides, where programs are link edited, it is important not to mix FORTRAN library modules from previous releases.

   a. If a math subroutine is in a library with a user subprogram, at link edit time, Release 16 modules will be edited with the incompatible math routine, and errors may occur due to the new linkages for error processing and extended save areas in some math routines.

   b. By replacing only some FORTRAN library modules of a load module, errors may occur in the linkage editor because of changed ENTRY names and/or at exe-

cution time because of internal linkage changes. There are now no reserved names due to FORTRAN library ENTRY names. The linkage editor maps will reflect this.

c. If a load module at a pre-release 16 level has boundary alignment errors, a Release 16 IHCADJST will be loaded from LINKLIB and errors will result.

2. There are two major considerations when a an installation changes the code in FORTRAN library modules.

a. Five of the library modules now exist in macro form with conditional assembly statements. Associated with each macro are two modules which will assemble the macro either with or without error message handling.

To change the module, change the macro then place it in MACLIB and assemble the module. The five modules are:

Macro	Module (without facility)	Module (with facility)
(1) IHCIBCOM	IHCFCOMH	IHCECOMH
(2) IHCFIOSM	IHCFIOSH	IHCEFIOS
(3) IHCDIOSM	IHCDIOSE	IHCEDIOS
(4) IHCTRACM	IHCTRCH	IHCETRCH
(5) IHCARITM	IHCFINTH	IHCEFNTH

Note that when these macros are assembled, only the expansions of the inner macros will be listed (not the original macro statement) and comments will be compressed.

b. The linkage to FIOCS# and to DIOCS# have been changed. The normal return point from an I/O call is +6 (+2 when an error is detected) from where Register 0 points upon entry to FIOCS# or DIOCS#.

```
Example:   L       REG1, ADFIOCS
           BALR    REG0, REG1
           DC      AL2(X)
           B       ERROR
GOODIO     ST      ...
```

3. The entry points IHCERRM and ERRMON for error processing are in the following modules:

```
IHCFCOME - E level library
IHCERRM  - G-H level library-error handling
IHCTRCH  - G-H level library-no error facility
```

4. There will be no significant effect on object time performance for error-free programs. However, additional processing time will be required to handle error conditions.

5. The storage requirements of the FORTRAN library will be affected by the addition of improved object time error messages resulting in an increase in the core storage requirements of an object program. This increase will be 150 bytes for FORTRAN E (400 bytes for FORTRAN G and H), plus an average of 200 bytes additional for each library subprogram called. In addition, if the user selects the optional features for FORTRAN G & H, core storage from 4500 to 5300 bytes will be required for an error monitor depending upon the action desired.

6. **The Programmers' Guides specify the defaults for the option table IHCUOPT.** Most of the errors will allow ten occurrences and five messages except:

a. IHC208I, IHC210I, IHC215I-unlimited errors.

b. IHC217I, IHC230I one error, one message and the entry for IHC230I cannot be dynamically changed.

c. The current buffer is printed for error numbers 212, 215, 218, 221-225.

d. When there is an END or ERR parameter for a READ, unlimited errors are allowed.

7. A program loop can occur when the table is set to allow unlimited errors.

8. Each installation can assemble an option table with its own default values and addresses of user exit routines by referring to the description of the SETENT and PREFACE macros in the Programmers' Guides. The error message facility can also be used to process user detected errors.

9. Message text and job execution changes.

   a. Without the facility, execution is the same. The messages are the same except now there is informational text with error messages 215, 216, 221-225 and from 241 on.

   b. With the error facility, execution will terminate after 10 occurrences of error number 207, 209 (previously unlimited overflows and divide checks were allowed). For other errors, execution will continue with standard fixup and the extended error message and traceback are printed. A message is printed stating if user or standard fixup is taken, and there is an error summary at the End of Job.

   c. The error number 210 is divided into four messages for the error facility.

IHC207I-overflow IHC209I-divide check
IHC208I-overflow IHC210I-boundary alignment

Additional planning information is being made available to each district SD & I Center.

OS/DOS/360 2420-7 TAPE SORT TIMINGS ON S/360-50H

OS/DOS/TOS/360 Support for the 2420-7 Magnetic Tape Units was announced in Program Announcement P68-16. It was stated that actual tests under OS and DOS on a Model 50 H indicated improvements of up to 31% compared to the 2401-6.

Exhibit 1 provides some additional detail on some actual tests made on the Model 50 H. The information is provided for planning purposes pending the availability of formal SRL documentation.

DOS/360 AL RE-OPENING SEQUENTIAL DISK FILES

The following field contribution discusses the re-opening of sequential disk files using DOS/360 Assembler Language with blocked records and/or two I/O areas.

A problem can occur if it is necessary to re-open a file within one job step.

The Supervisor and I/O Macro Manual (C24-3429-3 page 70) states that if further processing of a closed file is required at some later time in the program, the file must be opened again. The data will be incorrectly written after re-opening.

The CLOSE routine usually must write the last block as a short record, but the initial settings for blocking factor and I/O area pointer within the DTF table is not re-stored, therefore, its re-use is not possible within the same program.

Several alternatives are available to the programmer (which should be evaluated prior to implementation in the users own environment):

1) Define the file as unblocked with one I/O area, but that may significantly affect run time.

2) Define the file as a 'work' file. The programmer must then perform his own blocking and padding of the last block.

3) Set up the DTF table as a separate phase which could be LOADED from the Core Image Library before re-opening. This would require additional control cards for linkage editing.

4) Move the DTF table to a save area before the first OPEN and restore it before any re-openings. This requires additional coding by the programmer.

OS/DOS/360 2420-7 TAPE SORT TIMINGS ON S/360-50H

System/360 Model	Operating System	2816 SW	Number Work Units	Data Set Size	Record Size	Block		Core Storage Used	Time in Minutes Using	
						In/Out	Sort		2401-6	2420-7
Mod 50H	OS	No	3 on 2 ch.	300K	80	90	125	150K	34.8	26.4
		No	4 on 2 ch.	200K	200	35	50	150K	49.8	35.4
		No	4 on 2 ch.	125K	500	10	20	150K	81.0	55.8
		Yes	4 on 2 ch.	300K	80	90	125	150K	22.8	20.4
		Yes	3 on 2 ch.	200K	200	35	50	150K	48.6	37.2
		Yes	4 on 2 ch.	125K	500	10	20	150K	66.6	45.6
	DOS	No	4 on 2 ch.	300 K	20	350	500	150K	20.2	19.0
		No	4 on 2 ch.	300K	80	90	125	150K	31.5	26.0
		No	4 on 2 ch.	200K	200	35	50	150K	49.3	34.3
		Yes	4 on 2 ch.	300K	20	350	500	150K	19.9	19.5
		Yes	3 on 2 ch.	300K	80	90	125	150K	35.0	31.8
		Yes	3 on 2 ch.	200K	200	35	50	150K	47.6	37.0

Exhibit 1

DOS/360 RELEASE 16 QTAM SWITCHED LINE CONSIDERATIONS

The following considerations are for DOS/360 QTAM with switched lines.

Check the DTFQTs for line groups to be sure that:

1. INTVL is not specified. It is not flagged in the assembly, but results in ceasing to POLL after some polling has been done.

2. CPRI is not specified. It is not flagged in the assembly and could cause cessation of polling.

3. Lines using Auto-Call feature and lines not using it are in different DTFs. This is not checked in the assembly and may not cause problems at first. Eventually, when Auto-Call lines are all busy and QTAM is trying to get a line to send to a terminal, it will try to use one of the non-auto-call lines to call out on. The system will go into an abeyant state with no messages or indications of the problem with the system light still on. (NOT wait state).

4. TERM entries within a line group have EXACTLY the SAME FORMAT. This is not checked in the assembly. All TERM entries in one line group (on lines specified by one DTF) must have:

   a) Same option fields specified

   b) Same number of dial digits and access characters. Situation a) will show up immediately because the TERM entry with fewer option fields will not be addressed properly (wrong access characters location, including dial digits, if any) so contact will never be made from the CPU to this terminal. Proper contact will be made with longer TERM entry terminal. (This is a bug and is planned to be fixed).

      Situation b) will show up when trying to address a terminal after a CPU to terminal call connection has been made. The TERM phone will ring, dataset goes into mode, but no writing at the terminal occurs. Wrong addressing characters were picked up by QTAM.

These problems all have to do with ACLOC specifying an improper value. Be sure ACLOC is specified:

1. when any OPTIONS are used.

2. when Intercept is used in ENDSEND (you must use 3 byte option for Intercept)

3. when INTREL is used in OPCTL in RCVHDR (3 byte option required here too).

On TWX terminals TERM entires, be sure ID = is long enough, including ending sequence sent out by TWX terminal, usually XOFF character. QTAM generates a 2 set entry of the ID. On a Write operation, QTAM reads into the first set the ID sent from the terminal, including hardware ending sequenc, and does exact compare against the second set. If compare is unequal, no writing takes pace and no indication or message is provided to the user. Core can be examined to see if what was read in matches what was specified in the program (look at TERM entry), if the problem is recognized.

These and other "user" problems are listed in the last pages of a forthcoming QTAM Learner's Guide II. A preliminary copy is being sent to SD&I and TP representatives in the Field Systems Centers. The availability of the manual in its final form will be announced in a Publications Release Letter.

DOS/360 RELEASE 16 QTAM LOCAL 2260 POINTERS

Be sure to follow all the left margin lined items in TNL-N30-5019 to the DOS/360 Message Control SRL. Also, do not use EOB or EOBLC. Do not put any control characters in the message work area or buffer. When TYPEFLE=CMBND, all messages are expected to be conversational. That is, an operator is expected to type in START MI, text, then ENTER. QTAM will handle one message per 2848 at a time. It

reads it in, puts it to a process queue and waits to answer back from the process program. The RCVLPS uses mode converse, and cannot use mode initiate or process expedite. An answer is to be "put" from the process program for each message. The length of the write to the 2260L will be determined by the work area prefix length supplied by the user. Get message and Put message will be used. The workarea prefix will contain:

1. 2 bytes--length of data
2. 1 byte--X'02'
3. 1 byte--zero

Do not allow messages to stack up on disk. Multiple messages sent to a terminal will come out too fast for the operator to read. Mode converse will remedy this.

DOS/360 RELEASE 16 QTAM INTERCEPT/ RELEASE PROBLEM

The following (from SECOM) is provided for your information. No solution is available at this time. If you find a solution you might have it entered into SECOM through your Field Systems Center.

Some intercepted messages on destination queues will be sent on a release and some will not. This problem can be pin-pointed if the MCT field in destination QCB keeps showing up with the same number of unsent messages.

The sequence of problem events is as follows:

1. Messages are intercepted in Endsend.

2. The optional sub-field is filled in with the disk location of the first intercepted message, turning on the TERM intercept bit.

3. Release m is given later, turning off the intercept bit.

4. Before a release is affected (release m needs to have a new message prime the release) another intercept condition occurs and reinitializes the optional sub-field used by release m.

5. When another release m is affected, messages will be released from the new intercept disk location when a message is sent to this destination.

DOS/360 COBOL CROSS-REFERENCE LISTING

The District 12 Systems Design Center advises that it has available a program that will provide a cross-reference listing of a DOS/360 COBOL source deck. Requests should be sent to the Manager, Systems Design Center, 1728 Soo Line Building, 5th and Marquette, Minneapolis, Minnesota 55402. Requests should be made by IBM representatives only.

1287 OCR PLANNING AND INSTALLING INFORMATION

The following article provides planning and installing information for the 1287 OCR and includes references to other available information.

1) The first twelve of the 1287's on order have been installed successfully on schedule!

2 a) A copy of the "World of OCR" binder has been distributed to each Branch Office. In most Branch Offices it is being maintained by Systems Engineering. It contains practically all of the published technical material on the 1287 and twenty additional articles to assist in the successful installation of OCR equipment.

b) An OCR educational film has been provided to many Branch Offices. If your office does not have one and you have need of it, contact your Field Systems Center OCR representative or your Regional Product Marketing OCR representative.

c) The 1287 Installation/Marketing Maps were also made available to the Branch Offices. In the event that a copy is needed and is not available at your Branch Office, contact your Regional Product Marketing OCR representative.

3. The following is provided to you for planning purposes. A Type III program in development "IBM 1285/1287 OCR Programming Support Under OS/360" has been included in a SECOM weekly listing, dated March 28, 1968, sent to the Branch Offices. The SECOM retrieval code for the abstract (which includes the minimum system configuration) of this program as given in the listing is PROG-S0-3.7-001. All new Type III programs are announced when they become available from PID in the IBM Installation Newsletter. The program abstract is published and includes the necessary ordering information. No orders should be placed until the program is announced. The author states that the above program is planned for mid-1968 availability.

Changes to the quantity of the Newsletter shipped are to be made by the Administration Manager. (NOTE THAT EXTRA COPIES OF THE NEWSLETTER ARE NOT TO BE ORDERED FOR DISTRIBUTION OF INFORMATION TO CUSTOMERS.) He should update and send the DP Administration Managers' Document Control card (form number N44-780) to DP Document Control, DPD HQ.

Direct inquiries concerning the contents of this Newsletter to:

W. P. Gaffigan, Editor
IBM Installation Newsletter
Installation Planning-#713
DPD HQ
112 East Post Road
White Plains, New York 10601
Telephone 914 WH9-1900 Extension 4781

INFORMATION ABOUT THE NEWSLETTER

EVERY SALESMAN AND SYSTEMS ENGINEER SHOULD RECEIVE ONE COPY OF THE NEWSLETTER. Each FE Branch Manager should receive five copies for distribution to Customer Engineers.

The IBM Installation Newsletter is distributed only to IBM locations and is not available to customers. The Newsletter is separated into two parts. The first part contains information which is intended only for IBM personnel and the pages are labeled For IBM Internal Use Only. THESE PAGES, OR REPRODUCTIONS OF THEM, ARE NOT TO BE GIVEN TO CUSTOMERS. However, using your own good judgement, you may discuss pertinent information from these pages with a customer. The second part contains information which may be REPRODUCED by the Branch Office at their discretion and given to customers. These pages are not labeled with any restrictive classification. Note that when any abstracts for Type III programs are given to the customer, he must be informed that the disclaimer for Type III programs as contained in the Newsletter, applies. YOUR COOPERATION IN PROPERLY HANDLING THE CONTENT OF THE IBM INSTALLATION NEWSLETTER IS NECESSARY FOR ITS CONTINUED PUBLICATION.

NEW PUBLICATIONS ANNOUNCED BY PRL'S

The weekly PRL's (Publications Release Letters) are used to insure that all Salesmen and Systems Engineers are aware of new or revised Marketing Publications. Normally, each issue of the Newsletter will contain information from two PRL's, one following the other. The information will be placed in the Newsletter in its original form with no rearrangement of form numbers or titles. It is not intended to replace existing information and distribution sources. You should be certain that you are aware of these sources.

NOTE: Marketing Publications are obtained from The IBM Distribution Center Mechanicsburg, Pa.

The Publications Requisition, form number M02-0618-2, is used for ordering.

PRL #16 April 19, 1968

REFERENCE SOURCES

FORM NO.	TITLE	DISPOSITION
N20-1130.27	1130 SRL Newsletter Re: A26-5916-2	Scrap N20-1130.26
N20-1440.50	1440 SRL Newsletter Re: A24-3005-5	Scrap N20-1440.49

MARKETING PUBLICATIONS

221-0410-3	The IBM 29 Card Punch Proposal Insert	Use 221-0410-2
221-0484-0	The IBM System/360 Document Processing System Proposal Insert	NEW
221-0491-0	IBM System/360 MATLAN (MATrix LANguage) Proposal Insert	NEW
420-5035-1	IBM 6405 Payroll Application	Scrap 420-5035-0
520-1552-0	IBM S/360 Administrative Terminal System	NEW
570-0398-0	Information Recorder Model I Proposal Insert	NEW
570-0399-0	Information Reorder Model II Proposal Insert	NEW
A24-3081-3	Mark Read Station (Slanted Mark) for IBM 1418 and IBM 1428	Use A24-3081-2
A26-5914-5	IBM 1130 Physical Planning	Scrap A26-5914-4 & N26-0207
A27-3011-0	IBM 2760 Optical Image Unit -- Component Description	NEW
C24-3444-3	IBM S/360 Disk Operating System: Sort/Merge Program Specifications	Use C24-3444-2 & N28-2257
C24-3445-2	IBM 1401/1460 Timing Program for IBM S/360- Disk Operating System: Sort/Merge Program	Use C24-3445-1 & N24-5125
C24-5037-4	IBM S/360 DOS Supervisor & I/O Macros	Scrap C24-5037-3, N24-5306, N24-5285, N24-5329, N24-5344 & N24-5312
H20-0181-2	IBM S/360 APT Numerical Control Processor - Version 3 and 4 (360A-CN-10X) Application Description	Scrap H20-0181-1
H20-0494-0	Data Acquisition Multiprogramming System II Application Description Manual	NEW
H20-0509-0	IBM S/360 Administrative Terminal System - DOS (ATS/DOS) (360A-CX-18X) Terminal Operator's Manual	NEW
L22-6921-0	IBM S/360 Custom Feature Description, 2903 Special Control Unit Model 1 - RPQ 880836	NEW
N20-1080	TNL to Mathematical Programming System/360 (360A-CO-14X) Application Description Re: H20-0136-2	NEW
N20-1855	TNL to Mathematical Programming System/360 (360A-CO-14X) Read Communications Format (READCOMM) Program Reference Manual Re: H20-0372-0	NEW
N20-1856	TNL to Mathematical Programming System/360 (360A-CO-14X) Control Language User's Manual Re: H20-0290-2	NEW
N28-3013	IBM S/360 Time Sharing System Command Language User's Guide Re: C28-2001-1 with N28-3003	NEW
N28-3014	IBM S/360 Time Sharing System Operator's Guide Re: C28-2033-1	NEW
N30-5508	TNL - IBM S/360 Disk Operating System Basic Telecommunications Access Method - Re: C30-5001-4 with N30-5016, N30-5024, N30-5505, N30-5507	NEW
N33-9015	PL/I Subset Reference Manual Re: C28-8202-0	NEW
N33-9029	TNL to Mod 20 DPS Disk Sort/Merge Re: C26-3806-1, -2 with N33-9013, N33-9020, N33-9024	NEW
R20-8033-2	IBM S/360 Mod 20 Installation Programming - DASD, Program Information Manual	Scrap R20-8033-1
R20-9211-1	IBM S/360 Basic PL/I Coding - P.I. Course Description	Scrap R20-9211-0
X20-1760-1	Examples of Control Cards for Operating System/360 Reference Card	Scrap X20-1760-0
X22-6855-4	IBM S/360 Magnetic Tape Physical Planning Template	Scrap X22-6855-3
X22-6894-2	IBM S/360 Processors Models 25, 30 and 40 Template	Scrap X22-6894-1
Y20-0082-1	1800 Vehicular Traffic Control System (1800-UG-06X) Version 2 System Manual	Scrap Y20-0082-0
Y24-5100	TNL to IBM S/360 DOS Logical IOCS PLM Vol. 1 Re: Y24-5020-3	NEW
Y28-2039-1	IBM S/360 Time Sharing System Independent Utilities PLM	Scrap Y28-2039-0 & Y28-3060
Y28-3061-0	IBM S/360 Time Sharing System System Logic Summary PLM Re: Y28-2009-0 with Y28-3051	NEW
Y28-3065	IBM S/360 Time Sharing System Command Language PLM Re: Y28-2013-0 with Y28-3053	NEW
Y28-3066	IBM S/360 Time Sharing System Program Checkout System PLM Re: Y28-2014-0 with Y28-3054	NEW
Y28-3069	IBM S/360 Time Sharing System On-Line Test Control Program PLM Re: Y28-2042-1	NEW
Y28-3072	IBM S/360 Time Sharing System System Generation and Maintenance PLM Re: Y28-2015-1	NEW

PRL #17 April 26, 1968

C20-1691-0	Catalog of Programs for IBM System/360, Model 20 - March, 1968	NEW
N20-0030-19	Catalog of Programs for IBM System/360 - March 1968 Supplement to C20-1619-4	Scrap N20 0-18
N20-0031-15	Catalog of Programs for IBM 1130 Computer System and IBM 1800 Data and Control System - March, 1968 Supplement to C20-1630-3	Scrap N2 31-14
Z20-0305-17	KWIC Index to TIE March 1968 Catalog	Scrap Z20-0305-16 & Z20-1873
321-0012-0	IBM Systems Journal, Volume 6, Number 4	NEW
322-0039-0	IBM Journal of Research and Development, Volume 12, No. 2	NEW
520-2064-1	IBM System/360 Model 25 Facts Folder	Scrap 520-2064-0
542-0048-0	OCR Reference Chart	NEW
543-0047-0	Sighted Mod D Braille Instruction Manual	NEW
543-0048-0	Blind Mod D Braille Instruction Manual	NEW
570-0394-0	The Selection of A Career	NEW
570-0397-0	Information Recorder Concepts Brochure	NEW
570-0400-0	Information Recorder Application Sheet-Medical Insurance Claim	NEW
570-0405-0	Information Recorder Application - Hospital Clinical Data	NEW
570-0406-0	Information Recorder Application - Quality Control	NEW
570-0407-0	Information Recorder Application Sheet - Route Accounting	NEW
570-0412-0	Magnetic Tape 20 Microfilm	NEW
A24-3068-3	IBM 1401/1460 Misc. I/O Instructions	Use A24-3068-3 & N24-0393
C21-5001-0	BPS Utility Prog. for Distribution of DQS Specifications & Operating Guide	NEW
C24-3337-2	IBM S/360 Operating System RPG SRL Bulletin	Use C24-3337-1, N26-0515, N26-0530 & N26-0546
*		
C28-2010-1	IBM S/360 Time Sharing System System Generation and Maintenance	Scrap C28-2010-0 & N28-3009
C28-2025-1	IBM S/360 Time Sharing Systems FORTRAN Programmers Guide	Scrap C28-2025-0 & N28-3011
C28-2032-1	IBM S/360 Time Sharing System Assembler Programmer's Guide	Scrap C28-2032-0 & N28-3012
E20-0305-0	Food Service System for Hospitals	NEW
F20-0013-0	Direct Digital Dead Time Control	NEW
F20-0014-0	Adaptive Control	NEW
*C28-2043-2	IBM System/360 Time Sharing System Addendum	Scrap C28-2043-1
H20-0322-2	Attached Support Processor System (ASP) (360A-CX-15X) Version 2 Application Programmer's Manual	Scrap H20-0322-1
H20-0323-3	Attached Support Processor System (ASP) (360A-CX-15X) Version 2 System Programmer's Manual	Scrap H20-0323-2
H20-0335-1	1800 Vehicular Traffic Control System Version 2 (1800-UG-06X) Program Description Manual	Scrap H20-0335-0
H20-0508-0	IBM S/360 Administrative Terminal System - DOS (ATS/DOS) (360A-CX-18X) Program Description Manual	NEW
N20-1081	TNL to Data Acquisition Multiprogramming System (DAMPS) Version 2 Application Description Manual Re: H20-0494-0	NEW
N20-1086	TNL for H20-0365-0 Mechanism Design System - Gears & Springs	NEW
N20-1851	TNL to Mathematical Programming System/360 (360A-CO-14X) Application Description Re: H20-0136-2 with N20-1080	NEW
N20-1854	TNL to Administrative Terminal System/360 Application Description Re: H20-0297-0	NEW
N20-1857	TNL to Mathematical Programming System/360 (360A-CO-14X) Linear and Separable Programming - User's Manual Re: H20-0476-0	NEW
N20-1858	TNL to General Purpose Simulation System/360 Disk Operating System (360A-CS-19X) Operator's Manual Re: H20-0327-1	NEW
N23-0617	Education Newsletter Re: R20-1018-2	NEW
N27-3024	Changes to 2740/2741 Installation/Physical Planning Re: A24-3423-1 with N27-3018	NEW
N28-3007	IBM S/360 Time Sharing Systems IBM FORTRAN IV Re: C28-2007-1	NEW
N28-3015	IBM S/360 Time Sharing System Assembler Language Re: C28-2000-2 with N28-3000	NEW
N28-3016	IBM S/360 Time Sharing System Concepts and Facilities Re: C28-2003-2 with N28-3005	NEW
N28-3017	IBM S/360 Time Sharing System System Messages Re: C28-2037-1	NEW
N28-3018	IBM S/360 Time Sharing System System Programmer's Guide Re: C28-2008-0 with N28-3008	NEW
N28-3020	IBM S/360 Time Sharing System Assembler User Macro Instructions Re: C28-2004-1 with N28-3006	NEW
N33-9028	TNL to Mod 20 Disk Utility Programs Re: C26-3810-1	NEW
N33-9032	TNL to Mod 20 DPS/TPS RPG Re: C24-9001-3 with N33-9018 & N33-9025	NEW
R20-4114-0	IBM S/360 Operating System Concepts for Operators Learner's Guide	NEW
R20-8032-2	IBM S/360 Model 20 Installation Programming - DASD Education Guide	Scrap R20-8032-1 & N23-0011
R20-9075-2	IBM 85 Collator Operation and Wiring Programming Instruction Course Description	Use R20-9075-1

PRL #17 (Continued)

R20-9094-2	Basic Machine Operation Programming Instruction Course Description	Use R20-9094-1
R20-9111-1	FORTRAN for the IBM 1130 Programming Instruction Course Description	Scrap R20-9111-0
R20-9161-2	IBM S/360 Model 20 - Installation Programming - DASD Course Description	Scrap R20-9161-1
R29-0117-0	IBM S/360 PL/I Coding - How to Write a Basic PL/I Program Examinations	NEW
R29-0118-0	IBM S/360 Basic PL/I Coding Instructor Guide	NEW
R29-0119-0	IBM S/360 Basic PL/I Coding Case Study & Sample Solutions	NEW
Y20-0040-2	1130 Type Composition Program (1130-DP-04X) System Manual - Volume I	Use Y20-0040-1
Y21-0004-0	BPS Utility Program for Distribution of DOS PLM	NEW
Y28-2011-2	IBM S/360 Time Sharing System System Control Blocks PLM	Scrap Y28-2011-1
Y28-3063	IBM S/360 Time Sharing System Dynamic Loader PLM Re: Y28-2031-0	NEW
Y28-3068	IBM S/360 Time Sharing System FORTRAN IV PLM Re: Y28-2019-0 with Y28-3057	NEW
Y28-3070	IBM S/360 Time Sharing System System Service Routines PLM Re: Y28-2018 with Y28-3056	NEW
Y28-3071	IBM S/360 Time Sharing System Task Monitor PLM Re: Y28-2041-0	NEW
Z23-9121-1	1800 Data Acquisition - Control System Marketing Training Course Description	Use Z23-9121-0
Z32-0402-2	Computer Systems Training Part II (Sec. 5-9) Education Guide	Scrap Z32-0402-1
Z77-8033-0	Guidelines for Configuring FORTRAN Systems	NEW
Z77-8034-0	A Model for the Selection of Lock Boxes to Improve the Collection of Accounts Receivable	NEW
Z77-8035-0	Phase Overlaying: An Aid to Multiprogramming for Teleprocessing Application under DOS	NEW
Z77-8036-0	DOS COBOL Sequential Disk File Extension Using Checkpoint Restart	NEW
Z77-8037-0	Inventory Replenishment	NEW
Z77-8042-0	Debugging S/360 DOS COBOL Program Check Interruption	NEW

May 17, 1968 Issue No. 68-09

IBM INSTALLATION NEWSLETTER

OS/360 PARTIAL SUBSTITUTE FOR DISPLAY STATUS

The following technique was developed by an IBM customer and contributed to the Newsletter through an IBM representative. Potential users should evaluate its usefulness in their own environment prior to implementation. It has not been submitted to any formal IBM test.

For those OS/360 users wishing to bypass Display Status and still obtain certain information about tape files, the following may satisfy their needs and at the same time improve operator documentation.

Open module IGGØ19Ø verifies and creates standard label, output, tape data sets. By inserting the coding shown in Exhibit 1 immediately after creating HDR2, the operator will be notified via the console with the following message:

Jobname/Stepname Drive Number
Data Set Name Volume Serial

This then provides the operator with the same information as Display Status and it is also very beneficial in a multiprogramming environment.

The inclusion of the first compare nullifies the operator's notification of system data sets. (i. e., those data sets having A's in the name.)

By replacing the following instruction

MVC MSGTEXT (44), JFCBDSNM

at hexidecimal location ØØØ354 with

MVC MSGTEXT (11), FL1ID

the operator will also be notified, when tape retention checks occur, with the data set name which already exists on tape, rather than the name of the data set about to be written.

Naturally something had to be given up to obtain the above since module IGGØ29ØP is exactly 4Ø95 bytes. We found it necessary to delete the User Verification and the Password and Security Check coding routines to obtain enough bytes to include the above. It must be noted here, that this was our approach and that it is left up to the user's discretion as to what coding he wishes to replace.

OS/360 PARTIAL SUBSTITUTE FOR DISPLAY STATUS

```
          CLC   JFCBDSNM(3),AAA   AAA defined as DC CL3'AAA'
          BE    PASTCODE
          LA    Ø,8Ø
          BAL   1,*+4
          SVC   1Ø
          LR    11,1
          MVI   Ø(1),C'Ø'
          MVC   1(79,1)Ø(1)
          MVC   9(17,1),FL2JSID
          MVC   28(3,1),UCBNAME
          MVC   33(11,1),JFCBDSNM
          MVC   46(6,1),SRTEVOLI
          LA    Ø,62
          SLL   Ø,16
          ST    Ø,Ø(1)
          SVC   35
          LR    1,11
          LA    Ø,8Ø
          SVC   1Ø
PASTCODE  EQU   *
```

Exhibit 1

DOS/360 ZERO LENGTH RECORD WRITE ALTERATION

The following is a field contribution which has not been submitted to any formal IBM test. Potential users should first evaluate its usefulness in their own environment. No maintenance can be expected.

SDMODW gives a channel program error when writing a record of zero length (Applies to Release 16 and earlier).

SDMODW does not test for data length of zero in the chained data CCW when writing an EOF record (a count record with key length and data length fields zero).

SOLUTION: Test for record length of zero, and if so, do not use a chained data CCW in the channel program. In SDMODW routine IJGW-WRTE following MVC IJGWWCCW+1 (7), IJGWVGY+1 add the following 4 instructions:

```
CLC  IJGWCCW+6 (2), IJGWCERO
BNE  IJGWPAT1
MVI  IJGWWCCW+4, X'20'
B    IJGWPAT2
```

and add the labels to the next 2 instructions as:

```
IJGWPAT1  MVI  IJGWWCCW+4, X'AO'
IJGWPAT2  MVI  IJGWWCCW, X'1D'
```

This eliminates the chain data bit in IJGWWCCW.

DOS/360 BASIC PL/I EXTENDING IS FILES

The following is a field contribution which has not been submitted to any formal IBM test. Potential users should evalute its usefulness in their own environment.

In order to extend an existing Index Sequential file with data having keys that are all in sequence and higher than the highest key already in the file, the LODIS macro can be used with an ISE DLAB for that file.

The same program that loaded the file can be used to extend the file by using an ISE instead of an ISC DLAB.

The RUAIS macro should not be used for extending. Adding and extending are not the same concept. Extending requires an extension of the indexes also, and the add does not have this ability.

DOS/360 DISK SORT/MERGE ALTERATION FOR FOREGROUND EXECUTION

The following contribution has not been submitted to any formal IBM test. The author states that it has been subjected to limited testing. Potential users should evaluate its usefulness in their own environment prior to implementation. The approach applies to DOS/360 Release 16 (Version 3 Modification 0).

Any problems encountered (including possible solutions) can be reported by the local IBM representative to the Field Systems Center.

The following procedure can be used to create separate Disk Sort/Merge programs which can be executed in one or both foreground partitions (using a BJF Supervisor under DOS V3M0). This is a REP card approach and does not make the sort self-relocating. Sorts can be run concurrently in all three partitions, each sort being called by its own unique program name. Programming Temporary Fixes (PTF's) shipped by the IBM PID with DOS V3M0 (Release 16) do not conflict with the REP card addresses except as noted under step 4a and 4b Below).

1. Catalog the desired variant(s) of the sort/merge program into the CIL for background execution when needed.

2. Display and punch all relocatable modules associated with the sort. This can be done using the statements:

```
// JOB PCHPRT DISK-SORT RELO MOD-
ULES

// EXEC RSERV
   DSPCH IJO.ALL
/&
```

   Mark the decks and save them for later use. Some modules (IJOSM002, 003, 004, 005, 006, 008, 010, 501, 301, 302, 303, 304, and 402) will not be used for the foreground sort change but may be needed later to apply PTF's or other changes.

3. Select a unique phase naming convention for use with the sort program in the desired

foreground partition(s). The background sort uses phase names DSORTnnn, where nnn is a 3-digit suffix for each of the 26 sort/merge phases. Any or all of the first 5 characters should be changed (considering the FGP convention for applicability) and used in lieu of DSORT. For example, a frequently used F1 sort might use phase names starting with "FGPS1" whereas a less-used F2 sort might use "F2SRT" phase names. These same five-character names will be used in the EXEC statement to perform the respective foreground sorts.

4a. Punch the REP cards shown in Exhibit 1 to access LUB information for the pertinent partition. The first change provides access to system LUB data, while the last three cards allow accessing programmer LUB information. The ESID required in all REP cards being punched is 001.

4b. Punch REP cards to change the phase names fetched within the modules shown in Exhibit 2. The present phase name should be changed to agree with the naming convention previously selected. Since a common routine in the root phase is used to fetch many of the phases, only part of the modules contain names which need be changed.

4c. Punch the following REP cards to allow successful execution (without an addressing exception) above 32K. It is possible that other similar addressing problems may show up with extensive use of the sort in foreground partitions.

Module	Address	Data
IJOSM102	001950	5840
IJOSM102	001972	5040

5a. Insert the REP cards for the desired partition just ahead of the END cards for the affected modules, and then catalog the revised modules back into the relocatable library using the original module names. This eliminates the need to change INCLUDE statements in the Processor Generation Modules. The cataloging can be done using the statements shown in Exhibit 3.

5b. Replace all PHASE cards in the Processor Generation Modules shown in Exhibit 4 with similar cards having the revised phasenames, i.e., consistent with the REP cards punched at step 4b.

5c. Catalog the revised Processor Generation Modules back into the relocatable library using the statements shown in Exhibit 5.

6. In JBF mode with foreground partition, size properly allocated, link-edit the desired sort program variant for the specific partition. This can be done by using the following statements:

```
// JOB LINK EDIT DISK SORT
// OPTION CATAL
   ACTION F1 (or F2)
   INCLUDE IJOSM (or other pertinent
                  processor gen. module)
// EXEC LNKEDT
/&
```

7. Steps 5 and 6 should be repeated if the sort is also to be executed in a second foreground partition. It will probably be necessary to condense the relocatable library at some point during the cataloging operations since the revised modules use the same names as the equivalent background modules, the space originally occupied by the background modules is available.

When all cataloging operations have been completed, the sort modules can be deleted from the relocatable library. The relocatable decks should (obviously) be retained for back-up and maintenance needs.

DOS/360 DISK SORT/MERGE ALTERATION FOR FOREGROUND EXECUTION

	Module	Address	Data
For F2:	IJOSM001	0018C8	4881, 007C, 4838, 0022, 0700
	IJOSM007	001A7C	0002
	IJOSM009	00197A*	0002
	IJOSM104	001E3E	0002
For F1:	IJOSM001	0018C8	4881, 007C, 4838, 0032, 0700
	IJOSM007	001A7C	0003
	IJOSM009	00197A*	0003
	IJOSM104	001E3E	0003

**Address is 00198A if PTF 360N-00342 is applied.

Exhibit 1

DOS/360 DISK SORT/MERGE ALTERATION FOR FOREGROUND EXECUTION

Module	Address (V3M0)	Present Phase Name
IJOSM001	002D80	DSORT002
IJOSM101	001CC0	DSORT102
IJOSM102	001AA0	DSORT104
"	001AA8	DSORT103
IJOSM103	002300	DSORT501
"	002308	DSORT105
IJOSM104	002218	DSORT105
IJOSM105	002050	DSORT105
IJOSM201	001C10	DSORT501
"	002740	DSORT301
IJOSM202	002BC8	DSORT302
IJOSM203	001C80	DSORT501
"	002618 **	DSORT303
IJOSM204	002A48	DSORT304
IJOSM401	0024F8	DSORT402

**Address is 002620 if PTF 360N-00349 is applied.

Exhibit 2

DOS/360 DISK SORT/MERGE ALTERATION FOR FOREGROUND EXECUTION

```
// JOB CATALOG REVISED RELO MODULES
// EXEC MAINT
   CATALR IJOSM001
      (IJOSM001 Relocatable Deck including two REP cards)
   CATALR IJOSM007
      (Relo Deck with one REP card)
   CATALR IJOSM009
      (Relo. Deck with one REP card)
   CATALR IJOSM101
      (Relo. deck with one REP card)
   CATALR IJOSM102
      (Relo. deck with four REP cards)
   CATALR IJOSM103
      (Relo. deck with two REP cards)
   CATALR IJOSM104
      (Relo. deck with two REP cards)
   CATALR IJOSM105
      (Relo. deck with one REP card)
   CATALR IJOSM201
      (Relo. deck with two REP cards)
   CATALR IJOSM202
      (Relo. deck with one REP card)
   CATALR IJOSM203
      (Relo. deck with two REP cards)
   CATALR IJOSM204
      (Relo. deck with one REP card)
   CATALR IJOSM401
      (Relo. deck with one REP card)
/*
/&
```

Exhibit 3

DOS/360 DISK SORT/MERGE ALTERATION FOR FOREGROUND EXECUTION

Processor Generation Module	Present Phase Names
IJOSMPH0	DSORT, DSORT002, DSORT003, DSORT004, DSORT005, DSORT006, DSORT007, DSORT008, DSSORT009, DSORT101, DSORT501
IJOSMPH1	DSORT101, DSORT102, DSORT103, DSORT104, DSORT105
IJOSMF23	DSORT201, DSORT202, DSORT301, DSORT302
IJOSMV23	DSORT203, DSORT204, DSORT303, DSORT304
IJOSMPH4	DSORT401, DSORT402

Exhibit 4

DOS/360 DISK SORT/MERGE ALTERATION FOR FOREGROUND EXECUTION

```
// JOB CATALOG REVISED PROCESSOR GENERATION MODULES
// EXEC MAINT
   CATALR IJOSMPH0
      PHASE card (revised), INCLUDE card (unchanged), END card
   CATALR IJOSMPH1
     (similar to above)
   CATALR IJOSMF23
     (etc.)
   CATALR IJOSMV23
     (etc.)
   CATALR IJOSMPH4
     (etc.)
/*
/&
```

Exhibit 5

DOS/360 DISK SORT SRL DOCUMENTATION ERRORS

Some documentation errors exist in the DOS/360 System Generation and Maintenance manual (C24-5033, both -3 and -4). In the later revision, the errors are on page 222 of Appendix D:

a) The following two statements should be included in the Intermediate Processor Generation Module IJOSMPH0 (at the top of the page) rather than under IJOSMF23 near the bottom of the page:

```
PHASE      DSORT501
INCLUDE    IJOSM501
```

b) The Intemediate Processor Generation module shown as IJOSMERG (middle of page 222) and containing the following five statements) should be named IJOSMPH4:

```
PHASE DSORT401
INCLUDE IJOSM401
PHASE DSORT402
INCLUDE IJOSM402
END
```

These modules are correct in the relocatable library of DOS Release 16 (V3M0), so the error is strictly in documentation. However, the discrepancy could cause confusion and might lead to errors since PHASE cards in these Processor Generation Modules must be changed to implement the foreground sort changes.

The following Primary Processor Generation modules were inadvertently omitted from Release 16:

IJOSMFOM, IJOSMVOM, IJOSMFVS, IJOSMFOS, IJOSMVOS, and IJOSMERG.

Users can get these from the SYSGEN SRL. These modules will be included in a future Release of DOS/360.

S/360-20 INVALID CHARACTER CHECKING FOR INPUT CARDS

The following is a field contribution which has not been submitted to any formal IBM test. Potential users should evaluate its usefulness prior to implementation.

A simple Basic Assembler Language routine was written which issues a halt if an error occurs in a presumably valid field.

1. In the case of an MFCM Read, there are three TIOB instructions in the MFCM Routine. The instruction to be modified is approximately 250 hexidecimal bytes into the routine.

2. By including a subroutine in his RPG program, the user can realize the advantages of the built in error-checking while simultaneously reading and bypassing his invalid character(s).

In this program example, there are two types of cards which are known to contain invalid characters in card columns 55-56-57. The subroutine, therefore, does the error checking for columns 1 to 54 and 58 to 80. With simple programming modifications, this routine can be used to check any combination of columns within a particular card type(s).

The RPG Input and Calculation specifications are shown in Exhibit 1, and the BAL routine is shown in Exhibit 2.

S/360-20 INVALID CHARACTER CHECKING FOR INPUT CARDS

IBM INTERNATIONAL BUSINESS MACHINES CORPORATION

REPORT PROGRAM GENERATOR INPUT SPECIFICATIONS

3 4 5	6	7–14	15 16	17	18	19 20	21–24	25	26	27	28–31	32	33	34	35–43	44–47	48–51	52	53–58
01Ø	I	FILEIN	AA			Ø1	8Ø		C	1									
02Ø	I															1	8Ø		FLD1
03Ø	I															1	7		ACCTNO
04Ø	I															8	12		POLNO
05Ø	I															13	54		DESC
06Ø	I															7Ø	75	2	AMOUNT
07Ø	I		BB			Ø2	8Ø		C	2	79	N	C	3					
08Ø	I															1	8Ø		FLD2
09Ø	I															1	7		ACCTNO
10Ø	I															8	28		NAME1
11Ø	I															29	49		NAME2
12	I																		
13	I																		

IBM INTERNATIONAL BUSINESS MACHINES CORPORATION

REPORT PROGRAM GENERATOR CALCULATION SPF TIONS

3 4 5	6	7 8	9 10 11	12 13 14	15 16 17	18–27	28–32	33–42	43–48	49–51	52	53	54 55	56 57	58 59 60
01Ø	C		Ø1				EXIT	SUB							
02Ø	C		Ø2				EXIT	SUB							
03	C														
04	C														
05	C														
06	C														
07	C														
08	C														
09	C														
10	C														
11	C														

Exhibit 1

IBM IBM System/360 Basic Assembler Short Coding Form

PROGRAM | PUNCHING INSTRUCTIONS | GRAPHIC | PUNCH
PROGRAMMER | DATE

Name	Operation	Operand	STATEMENT
SUB	START	0	
	EXTRN	FLD1	
	EXTRN	FLD2	
	EXTRN	IN01	
	EXTRN	IN02	
	BASR	15,0	
	USING	*,15	
	BC	15,PROC	
ADR1	DC	Y(FLD1)	
ADR2	DC	Y(FLD2)	
YIN1	DC	Y(IN01)	
YIN2	DC	Y(IN02)	
WD4	DS	H	
TABL	DC	X'FF000000000000000'	
	DC	XL16'00'	
	DC	XL16'00'	
	DC	XL16'00'	
	DC	XL16'00'	
	DC	XL16'00'	
	DC	XL16'00'	
	DC	XL16'00'	
	DC	XL16'00'	
	DC	XL16'00'	
	DC	XL16'00'	

A standard card form, electro number 6507 is available for punching source stateme

IBM IBM System/360 Basic Asse

PROGRAM | PUNCHING INSTRUCTIONS | GRAPHIC | PUNCH
PROGRAMMER | DATE

Name	Operation	Operand	STATEMENT
	DC	XL16'00'	
	DC	XL16'00'	
	DC	XL16'00'	
	DC	XL16'00'	
	DC	XL16'00'	
	DC	XL8'00'	
CON1	DC	XL16'00'	
	DC	XL16'00'	
	DC	XL16'00'	
	DC	XL6'00'	
CON2	DC	XL16'00'	
	DC	XL7'00'	
PROC	STH	11,WD4	
	SR	11,11	
	LH	11,YIN1	
	CLI	0(11),X'F0'	
	BC	6,TRY	
	SR	11,11	
	LH	11,ADR1	
	BC	15,TRN1	
TRY	SR	11,11	
	LH	11,YIN2	
	CLI	0(11),X'F0'	
	BC	6,STOP	

A standard card form, electro number 6507 is available for punching source statem

Exhibit 2

S/360-20 INVALID CHARACTER CHECKING FOR INPUT CARDS

ic

PROGRAM | PUNCHING INSTRUCTIONS | GRAPHIC
PROGRAMMER | DATE | PUNCH

STATEMENT

Name (25–30)	Operation (32–36)	Operand (38–45)	(50)
	SR	11,11	
	LH	11,ADR2	
TRN1	TR	ϕ(54,11),TABL	
	CLC	ϕ(54,11),CON1	
	BC	8,TRN2	
	DC	X'99ϕϕϕ1ϕ1'	
TRN2	TR	57(23,11),TABL	
	CLC	57(23,11),CON2	
	BC	8,STOP	
	DC	X'99ϕϕϕ1ϕ1'	
STOP	SR	11,11	
	LH	11,WD4	
	BCR	15,14	
	END	SUB	

Exhibit 2 (Continued)

1130 - CSP HIGH PERFORMANCE ALTERATION OF GET

The following routine has not been submitted to any formal IBM test. Potential users should evaluate its usefulness in their own environment prior to implementation.

Version 3 of the Commercial Subroutine Package, just announced, will be available June 1, 1968. In addition to supporting the 1403, 2501, and 1442-5, all the character handling (GET, PUT, EDIT, MOVE, FILL etc.) and the Decimal Arithmetic (ADD, SUB. MPY, DIV, etc.) subroutines have been recoded from FORTRAN to Assembler. This revision caused them to shrink in size (to about 0.6 their former size, on the average) and to run considerably faster.

Except for GET and PUT, which still use floating-point arithmetic, the new routines are several times faster than the old ones. The speed improvement factor for GET and PUT is approximately 1.5. A high-performance variant of the GET routine (called GIT) which will improve the factor of 1.5 is shown in Exhibit 1.

It operates in a manner, almost identical to GET, with the exception that the shift parameter is ignored (or assumed to be 1.0, whichever way you choose to look at it). Because SHIFT is, and should be, 1.0 in most cases, this should not be a serious impediment to the use of GIT. If a SHIFT different than 1.0

```
VAR = GET (KOL, 12, 18, .01)
```

is really needed, it should be just as easy to code

```
VAR = GIT (KOL, 12, 18, 1.0) *.01
```

instead.

The improvement to be obtained in using GIT is shown in the summary below:

	Core Required (Approximate)	Average time per character, milliseconds, (Approximate)
GET, Version 2	148 words	3.33
GET, Version 3	98 words	2.50
GIT	92 words	.25

In addition, GIT used none of the real (floating point) arithmetic routines, and is consequently independent of precision--either standard or extended may be used.

A little arithmetic demonstrates the effect of the time saved. Consider, for example, a card to printer, job, with GET used to convert five ten-column fields. Suppose it now runs at 60 lines per minute, for a basic cycle length of 1 second or 1000 milliseconds. If GIT is used

in place of the Version 2 GET, the cycle is reduced by 50 *(3.33-.25) or 154 milliseconds, from 1000 milliseconds to 846. In terms of lines per minute, the improvement is from 60 lpm to 71 lpm.

At the higher speeds of the 1403, the potential improvements are even more impressive. At 200 lpm (a 300 milliseconds cycle), a saving of 154 milliseconds will increase the throughput to 410 lpm.

The calling sequence of GIT has been kept identical to that for the GET routine

```
GIT (KARD, JSTRT, JFIN,  SHIFT)
```

to permit easy conversion of source program. However, since the SHIFT parameter is ignored, a simple change will permit the call to reduce to

```
GIT (KAR, JSTRT, JFIN)
```

To implement this alteration, change the 16th source statement from

```
MDX  1  4
```

to

```
MDX  1  3
```

To avoid changing any source programs, the name of GIT can be changed to GET, and stored on the disk in place of GET. Change GIT to GET everywhere that name appears in the source program (1st, 2nd, 4th, and 10th cards), punch the cards, then assemble.

2314 DRAWER CLOSING OPERATING TIP

The operator can be assured that the 2314 drawer is properly closed by, after closing, attempting to open the drawer with a downward pressure of the handle. If the drawer has been closed properly, it cannot be opened without lifting on the handle.

1130 - CSP HIGH PERFORMANCE ALTERATION OF GET

```
        ENT     GIT
GIT     DC      *-*
        STX   1 FIN+1
        LDX   I1 GIT
        LD    1 0
TWO     S     I1 2
        STO     JCRD2+1
        STO     MISS+1
        LD    I1 2
        STO     GIT
ONE     S     I1 1
        A       ONE+1
        BSC     +
        LD      ONE+1
        STO     CNT+1
        MDX   1 4
        STX   1 DONE1+1
        LDD     ANS
        SRT     32
        STD     ANS
CNT     LDX   L1 *-*
JCRD2   LD    L1 *-*
        SLA     4
        SRA     12
        SRT     16
        STD     TEMP
        SLT     16
        S       NINE
        BSC   L ERR,-Z
MISS    LD    L1 *-*
        BSC   L MAYBE,+Z
        S       BLANK
        BSC   L ERR,Z
        BSC   L CONT
MAYBE   SLA     3
        BSC   L CONT,C
        STX   1 IR1
        LD      IR1
        S       ONE1
        BSC   L ERR,-Z
        SLA     1
        BSC   L CONT,C
        LD      ZIP
        STO     CNT+1
CONT    LDD     ANS
        SLT     2
        AD      ANS
        SLT     1
        AD      TEMP
        STD     ANS
        MDX   1 -1
        MDX     JCRD2
        STD   3 126
        LD      D159
        STO   3 125
        LIBF    NORM
        LD      CNT+1
        BSC   L FIN,-Z
        LIBF    SNR
FIN     LDX   L1 *-*
DONE1   BSC   L  *-*

ERR     SLT     32
        STD   3 126
        STO   3 125
        MDX     FIN

D159    DC      159
BLANK   DC      /4040
NINE    DC      9
TEMP    BSS   E 2
ANS     BSS     2
ZIP     DC      0
IR1     DC      *-*
ONE1    DC      1
        END
```

8-4 -19- 3

Exhibit 1

NEW TYPE III PROGRAMS *

The following are the abstracts of programs which have been recently made available from the Type III library.

The program, along with its complete abstract, will be incorporated in subsequent issues of the Catalog of Programs.

Programs may be obtained by submitting a properly completed "General Program Request Card" (Form Number 120-1145-1) to the Program Information Department, 40 Saw Mill River Road, Hawthorne, New York, 10532.

These programs and their related documentation are distributed by IBM in the author's original form and have not been subjected to any formal testing.

Any discussions of Type III programs with customers must emphasize the following points

1. Type III (IBM employee--contributed programs) are provided by the IBM company as part of its service to customers. They are not a part of the IBM product line support.

2. Type III programs have not been subjected to any formal test.

3. Recipients of Type III programs are expected to make the final evaluation as to the usefulness of the programs in their own environment.

4. There is no committed maintenance for Type III programs. However, any changes the author chooses to make will be announced in subsequent issues of the Catalogs of Programs.

5. IBM makes no warranty, expressed or implied, as to the documentation, function, or performance of Type III programs.

*NOTE: THE CUSTOMER MUST BE INFORMED THAT THE ABOVE APPLIES TO ANY OF THE FOLLOWING TYPE III ABSTRACTS FURNISHED TO HIM.

IBM OPERATING SYSTEM/360 BRAILLE UTILITY PROGRAM. The System/360 Operating System Braille Utility is a general purpose program designed to translate cards, magnetic tape, and direct-access input into Grade 1 Braille on an IBM 1403 Printer.

The program is also designed to translate any IBM SRL manuals that are contained on TEXT 90 Print tapes into Braille.

Configuration requirements are any model CPU that can accommodate Operating System/360, plus a 1403 Printer, 2540 Card Reader/Punch, and a 1052 Console. Either 9 track or 7 track tapes may be used. The Data Conversion feature is required for 7 track tapes.

Source Language - ALP.

Ordering Procedure: Order File Number 360D-01.0.005.

The program material can be obtained in card form or on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

If not specified, a 9 track DTR will be provided.

No tape submittal is required - the DTR will be provided by the Library.

FILING AND SOURCE DATA ENTRY TECHNIQUES FOR EASIER RETRIEVAL (FASTER-OS VERSION. FASTER (S/360 OS) is a programming system for retrieval and maintenance of IS files from terminals. The methods used are applicable to a wide variety of applications. FASTER incorporates a macro-language which provides for writing message processing programs on a functional level. Some of the major functions supported by the language include: modification and addition to and retrieval from indexed sequential files, data manipulation (including Boolean Logic), formatting of responses to selected terminals (including paging capability), message switching and audit data logging. The ability to background test a user system is also provided.

Using BTAM, FASTER provides line control support macros of a communication network consisting of IBM 1050, 2740 typewriter terminals and the IBM 2260 display terminal. Written in BAL, FASTER operates as a job in a single high priority partition under OS with a 65K minimum storage.

Ordering Procedure: Order File Number 360D-06.7.011.

To obtain program material, submit one 2400 foot reel of magnetic tape. Specify whether 7 or 9 track recording is required. If not specified, 9 track recording will be used.

The required tape may be ordered from IBM or supplied with your request for the program.

HOSPITAL SELECTIVE MENU PROCESSING PROGRAM. The Selective Menu Processing Program is applicable to the dietary departments of most hospitals. The program generates a dietetically suitable menu for each patient, for each meal, every day. The selective menus (menus from which patients select food items) are printed by the program on preprinted 1232 data recording sheets. Patients mark their choices on these sheets, and the sheets are returned to the data processing area. The program processes these punched cards and generates listings containing the patients food selections. The program process these punched cards and generates listings containing the total food requirements for each food preparation center in the hospital.

Selective Menu Processing Program features include:

a. Patient diet order maintenance and control.
b. Diet suitability checking.
c. Generation of pre-identified selective menus for each patient.
d. Tally of food requirements for each food preparation center each day.
e. Overall food requirements tally for entire hospital each day.
f. Capability of operating in a multi-hospital environment.
g. Hospital profile data set containing information and program parameters unique to each hospital.

Selective Menu Processing Program advantages include:

a. Increased patient satisfaction.
b. Improved patient diet control.
c. Reduced clerical costs.
d. Reduced plate waste.
e. More accurate planning of production. requirements.
f. Savings in supervisory time.

The required machine configuration is a System/360-30 Model E (32K bytes of storage) processing unit, 3-2311 disk drives, a 1232 Optical Mark Page Reader, and other standard peripheral equipment.

The programming system used is DOS.

The source language is COBOL.

Ordering Procedure: Order File Number 360D-13.0.002.

The program material can be obtained in card form or on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

If not specified, a 9 track DTR will be provided. No tape submittal is required. The DTR will be provided by the Library.

S/360 DISPLAY, ACCEPT AND ALARM MACROS FOR DOS ASSEMBLY LANGUAGE. DISPLAY, ACCEPT and ALARM are three macros which provide the DOS and TOS Assembler programmer with the ability to easily read data from the console, to write limited amounts of data on the 1052 console typewriter, printer and punch, and to activate the 1052 alarm. These macros are particularly useful for program debugging, for input of control information, and for output of small amounts of data. They use PIOCS and a common channel program, thereby using a minimum amount of core. Moreover, they are completely self-relocating.

Minimum machine configuration : machine configuration required for operation of the user's DOS or TOS plus decimal arithmetic. ALARM requires a Model 40 or above.

Source Language is ALP.

Ordering Procedure: Order File Number 360D-03.7.018.

This program material can be obtained in card form or on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

If not specified, a 9 track DTR will be provided.

No tape submittal is required - the DTR will be provided by the Library.

IBM SYSTEM/360 BRAILLE UTILITY PROGRAM. The IBM System/360 Braille Utility Program is designed to translate its input into readable Grade 1 Braille on a standard IBM 1403 Printer. Braille is a touch reading system used by blind individuals consisting of a configuration of dots in a 2 by 3 matrix.

This program consists of four parts: (1) BPS/360 Absolute loader (program no. 360 P-UT-017) (2) System control cards (distinguished by 2 slashes in card columns one and two (3) Object program cards constituting the braille program proper (4) A source deck for the System/360 braille utility program. This will be used to place the program in a BOS or DOS library. The source language is System /360 assembler language.

Requirements: Floating-point feature is not required. The minimum configuration is System/360 Model 30 with at least 16K positions of memory. The system must have a card reader, one tape drive (the Data Conversion feature is required for 7 track tapes) if dumps are to be requested, operator's console (1052) and 1403 printer. List from any input device except direct access.

Ordering Procedure: Order File Number 360D-01.0.006.

The program material can be obtained in card form or on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

If not specified, a 9 track DTR will be provided. No tape submittal is required. The DTR will be provided by the Library.

S/360 MANAGEMENT INFORMATION SYSTEM (MIS). The S/360 Management Information System (MIS) provides a terminal oriented capability to retrieve pre-defined reports, or subsets of them, from a single file using an easy-to-learn vocabulary. Key restrictions are: limited file maintenance capability; user normally reloads entire master file on a periodic basis; if sorting required, maximum is 300 records, 2260 support is local only.

Minimum machine configuration is 256K, Model 40: this is not recommended, since only 20-30K remain for background programs. In addition to normal S/360 Operating System requirements, the user should have 1-2311 and a 2848 Control Unit Model III for 2260 local operations and/or a 2702 for 1050, 2740, and 2741 terminals. Feature level details are specified in the documentation.

Storage requirements: one 2311 is recommended.

Source Language: with the exception of two modules (MIS440, MIS460) which are coded in COBOL-F, S/360 MIS is written in Assembler-F Language.

Ordering Procedure: Order File Number 360D-06.7.009.

To obtain program material, submit one 2400 foot reel of magnetic tape. Specify whether 7 or 9 track recording is required. If not specified, 9 track recording will be used.

The required tape may be ordered from IBM or supplied with your request for the program.

FILING AND SOURCE DATA ENTRY TECHNIQUES FOR EASIER RETRIEVAL (FASTER)-DOS VERSION. FASTER (S/360 DOS) is a programming system for retrieval and maintenance of IS files from terminals. The methods used are applicable to a wide variety of applications. FASTER incorporates a macro-language which provides for writing message processing programs on a functional level. Some of the major functions supported by the language include: modification and addition to and retrieval from indexed sequential files, data manipulation (including Boolean Logic), formatting of responses to selected terminals (including paging capability), message switching and audit data logging. The ability to background test a user sytem is also provided.

Using BTAM, FASTER provides line control support macros for a communication network consisting of IBM 1050, 2740 typewriter terminals and the IBM 2260 display terminal. Written in BAL, FASTER operates as a S/360 DOS foreground job with a 65K minimum storage.

Ordering Procedure: Order File Number 360D-06.7.012.

To obtain program magnetic submit one 2400 foot reel of magnetic tape. Specify whether 7 or 9 track recording is required. If not specified, 9 track recording will be used.

The required tape may be ordered from IBM or supplied with your request for the program.

GENERALIZED EXTRACTION PROGRAM--1410/7010. The GENERALIZED EXTRACTION PROGRAM--is a self-modifying utility program, written for the 1410/7010 Operating System, which will extract or delete records from any tape file based on the results of table searching and/or range testing in accordance with specifications submitted through the use of control cards. The output file can be in the same format as the input data or it can be reformatted and, in either case, additional constant information may be placed into it. Additional options permit the user to generate control totals on input and output, control output blocking factor, and to specify processing limits and random selection of input data. The program will generate input and output record counts, and output file nines padding if required.

The Generalized Extraction Program--1410/7010, GENEXT, requires an IBM 1410 system with 80,000 positions of core storage, three 729 tape units in addition to 1410 Operating System requirements.

Source Language - Autocoder

Ordering Procedure: Order File Number 1410-02.4.002.

The program material can be obtained on one 7 track Distribution Tape Reel (DTR).

No tape submittal is required - the DTR will be provided by the Library.

IBM 1440 CORE PUNCH PROGRAM. This program is used in conjunction with the existing 1440 object programs in order to create a new object deck which incorporates patches to the original deck. The program also resequences the new deck. The prime use of this program is to provide better back up for those people who have patched decks without source decks. What happens when the operator drops the payroll deck? Can it be put together again? Source language is 1440 Autocoder on Disk. The program requires 300 positions of core. The program was written to reside in the upper 300 positions of an eight (8) K memory. Minimum changes are required to alter the location. Machine: 1440 CPU (300 positions) and 1442 reader punch.

Ordering Procedure: Order File Number 1440-02.9.005.

Distribution will be in card form only.

IBM 1800 FORTRAN FLOWCHART. An IBM 1800/1130 FORTRAN program operating under Version 3 of TSX or under 1130 Monitor (with slight modification) accepts all valid IBM 1800/1130 FORTRAN Source statements to draw a flowchart on the IBM 1143/1132 printer. The program lists all non-executable statements and draws a unique block containing the full statement for each type of executable statement. The program connects the blocks where possible, draws up to 25 nested DO loops, prints state-

ment numbers, card numbers, and page numbers. Provisions are made for page headings. The output will differentiate between IBM 1800 process I/O and other I/O. After completing the flowchart, a sorted statement number versus page number index is printed. Provisions are made for stacked jobs. 1800 version under TSX required 16K, one disk, 1442, 1443. 1130 Version requires 8K, one disk, 1442, 1132. Programmed in FORTRAN.

Ordering Procedure: Order File Number 1800-00.2.001.

Distribution will be in card form only.

1800 DDC-TSX, A TIME-SHARING DIRECT DIGITAL PROCESS CONTROL PROGRAM. This program allows the 1800 Control and Data Acquisition System to replace conventional analog controllers in control of a process, i.e., perform direct digital process control. Operator communication is also implemented.

The IBM 1800 Time-Sharing Executive System (1800-OS-001, version 3), 24k of core storage, one 1053 printer, one 2310 disk file, and several RPQ's including the process operator's console (1892-10), are required. Written in symbolic assembly language, the program runs under control of the TSX System allowing the user to maximize the usage of the computer and have access to the complete programming system provided by TSX.

Ordering Procedure: Order File Number 1800-23.5.005.

The basic program material will be distributed in card form only.

The optional material consists of source code, flowcharts and listings distributed on one 2400 foot reel of magnetic tape, 9 track or 7 track (Data Conversion feature required). If not specified, 9 track recording will be provided. Submit one reel of magnetic tape. The required tape may be ordered from IBM or supplied with the request for the program.

TIMER SUBROUTINE FOR IBM 1800 TSX PROGRAM SYTEM. This timer subroutine replaces the standard 1800 TSX timer subroutine and expands the hardware timer A to 16 programmed timers A 0-A 15. Using the standard TSX timer calling sequence timing A 0 or Timer B are affected. In addition timers A 1 - A 15 are operating independent from each other and may be used for single or repeated subroutine calls in specified time intervals. A patch to interval timer control program will avoid stopping and resetting hardware timer A.

This timer program enables the user of 1800 TSX System to control the operation of a great number of customer equipments with different data rates and different on-off status. The program allows to supply a separate timer to each equipment and operate it in any combination with high flexibility.

Machine Configuration Requirements are the same as for IBM 1800 TSX System Version 3. Core storage required: 366 words. Source language is ASSEMBLER.

Ordering Procedure: Order File Number 1800-05.0.001.

Distribution will be in card form only.

IBM Installation Newsletter

July 26, 1968

Issue No. 68-14

CONTENTS

(G means Green Section)

Distribution: Branch Offices - DP Management, Salesmen, Systems Engineers, FE Managers. Regions, Districts, Education Centers, Field Systems Centers, Federal Systems Center, FE Area Offices, DPD HQ, FED HQ, WTC.

*Requires Immediate Attention

For IBM Internal Use Only

DPD HQ, White Plains, N. Y.

SELECTED SHORT SUBJECTS

The purpose of Selected Short Subjects is to bring together and highlight concise, factual and timely information which will indicate that action is to be taken by the IBM representative whose accounts are affected.

1. IBM Type I Program Maintenance

The May 1968 revision of page 2 of the Programming Section of the Sales Manual clarified the current operating procedures on program maintenance. The following information is extracted from Page 2 under Program Maintenance to be certain that you are aware of the current operating procedure.

1. Field Engineering has the responsibility to provide trouble analysis and to report it via an APAR, regardless of the program level actually in use.

2. A release is considered current from its initial distribution by PID until 30 calendar days after the availability of the next maintenance release. (e. g., Release 14 as modified by the Compiler-maintenance release is the current release and will remain so until 30 days after the initial availability of Release 15/16 from PID).

3. Corrections for valid program errors, permanent or temporary (PTFs), are tested on the release current at the time the correction is made available. A customer engineer when requested, will attempt to apply specific PTFs to a back level release. If it is not successful it is the customer's responsibility to retrofit the correction to his release level and validate its accuracy.

The information from the Sales Manual should be reviewed with Field Engineering and the Customer where there are any questions concerning IBM's Type I Program maintenance policy.

2. S/360 2916 LONG LINE ADAPTER RPQ

If you have a S/360 Model 50, 65, 67 or 75 installed or on order and need the capability to extend the channel interface up to two miles to accommodate remote control units or facilitate inter-system communication via the channel-to-channel adapter, you should read the article with the above title in this issue of the Newsletter.

3. 1130 - Commercial Subroutine Package (CSP), Version 3 SECOM Entries

Version 3 of CSP was released to the field in early June, 1968. Since then, several entries have been placed into the SECOM system.

Systems Engineers working with CSP users should keep up to date on these and any other future SECOM items. This is particularly true in light of the heavy customer utilization of the various CSP routines.

OS/360 RELEASE 16 SYSTEM GENERATION POINTER

The following information is provided for planning purposes only. The actual availability and specifications of all items will be as provided through the official IBM procedures, the Program Announcement and formal documentation.

The TRNMODE parameter has been deleted from the IOCONTRL system generation macro. Users whose decks contain this parameter should delete it before the SYSGEN of the Release 15/16 system. Inclusion of this parameter will result in the emission of a diagnostic and no output being produced by Stage I.

OS/360 RELEASE 15/16 LINKAGE EDITOR F-V2 Planning Information

The following information is provided for planning purposes only. The actual availability and specifications of all items will be as announced through the official IBM procedures, the Program Announcement and formal documentation.

Size Parameter

Invalid specification of the new Linkage Editor F SIZE Parameter can result in impaired editing

performance or, in extreme cases, in completely bypassing the editor output functions. The SIZE Parameter is intended to allow performance optimization by using available storage above the design point to expand the editor's internal tables and buffers. This use will reduce intermediate input/output. This extended storage allocation is controlled by the two operands of the SIZE Paramter. The first operand (Value 1) specifies the extent of storage to be assigned to the editor. The second (Value 2) designates the amount of Value 1 allocation to be used as input text buffer. If the buffer area is not large enough, overflow will be written on SYSUT1.

The editor space allocation routine will reconcile the SIZE values, the system space available, and the editor space requirements in accordance with the following criteria:

1. If SIZE is not included as an execute card parameter, default size values are assumed. These have been established at System Generation Time (SYSGEN).

2. The SIZE Parameter on an Execute card will override the SYSGEN'ed values for that Execute step.

3. When the available storage is less than that requested for Value 1, both Value 1 and 2 will be adjusted downward automatically by the Editor's space allocator. The minimum for Value 1 is the design point of the Editor involved. After determining a Value 1, the space allocation routine will attempt to satisfy the Value 2 request. If there is not sufficient space other buffers and tables will be adjusted downward to their minimums before Value 2 is reduced. Value 2 will then be adjusted to its minimum if necessary.

Note: SIZE must be used with caution when blocking SYSLIN and/or SYSPRINT data sets at more than the minimum 5 to 1. If blocking is specified at more than minimum, and if Value 2 is adjusted downward, NO output will be produced by the linkage editor.

SIZE must be specified consistent with the system conditions existing at editor execution time. Value 1 should not request more storage than the system will be able to supply to a GETMAIN (Region, partition, or total block of storage in PCP). Value 2 should not be too large to be directly allocated from the Value 1 area. If Value 2 (Text Buffer) takes a disproportionate amount from Value 1, the various tables and buffers will be reduced inordinately and the advantage of the greater text buffer will be lost to inefficient processing in the other data areas.

The final Value 2 (Text Buffer) can have an effect on the SYSLMOD record size written by the linkage editor. This output record size will be the smaller of the device type record size, or one-half of the effective Value 2. This factor must be considered when greater than 3K records are desired.

Recommended procedure for the use of the Size option involves two stages of specification: 1. At SYSGEN; and 2. At Editor execution as an EXECUTE card parameter.

The sizes specified in the EDITOR macro for System Generation should be consistent with the configuration of the intended system and its normal use. The generated default Value 1 should represent the minimum amount of storage commonly used on those occasions when individual operating conditions call for the expansion (or contraction) of the values for optimum operation.

Value 1 should not exceed the actual storage area which will be available. Value 2 should be consistent with its Value 1 by not forcing any rounding down in the allocation of areas.

Exhibit 1 shows the relationships between minimum Value 1 and maximum Value 2. This relationship should be preserved in that any increase to Value 2 should be matched in Value 1.

Treatment of Named Common

References to NAMED COMMON areas with the same name are collected and modified to relect the longest length. After data has been received for the common, the length should not be increased by subsequent references to the same NAMED COMMON. If an increase is attempted, an incomplete load module may be

created. The following four conditions must be satisfied for this incomplete load module possibility to occur:

1. The track size of the output device is large compared to the size of CSECTS being processed.

2. More than one CSECT contains NAMED COMMON of the same name.

3. The NAMED COMMON contains data.

4. A NAMED COMMON processed subsequent to the initializing module contains a length greater than any NAMED COMMON of the same name previously processed.

This does not represent a new condition for F level linkage editors. (In order to create the most efficient load module, as many CSECTS as possible are grouped into one output record.) In order to avoid this NAMED COMMON situation, the grouping concept would be nullified at a cost of increased secondary storage space, and increased FETCH overhead.

The exposure of this item is expected to be nominal. It was mentioned only once during the life of the replaced F Linkage Editor and the current Editor SRL publications clearly describe the problem. Also the USASI FORTRAN Standard states:

The sizes of labeled common blocks with the same label in the program units that comprise an executable program must be the same.

DOS/360 V-III ISAM PRIME DATA AREA FULL CONDITION

ISAM sets error bits in a field known as "filename C' for certain types of I/O conditions resulting from its file handling. One such bit, bit 2 was set on in Version II if the prime data area was full. In Version III, this same bit is set to signify "prime data area full" but the setting occurs on starting to write on the last cylinder of the Prime Data area. This means that the user will sense this condition in Version III with a full cylinder of data area still available.

This change in Release 16 was originally made to satisfy the ISAM recovery problem of having room left in the ISAM file to write an end-of-file marker on a cancel condition. Since one full cylinder was more than enough room for this situation, Release 17 was changed in ISMOD9 to compare for the "Prime Data Area Full" condition on both cylinder and track so that bit 2 is set only at the start of the last track in the last prime data area cylinder.

However, this check is not logically valid if independent overflow tracks are assigned on the last Prime Data cylinder. For instance, track 7 could be the true last Prime Data track but ISAM will check only for track 9.

Latest information indicates that ISAM may be changed again in Release 18 to check for the beginning of the true last Prime Data area track. The actual availability will be as provided through the formal IBM procedures and documentation.

If using Release17, you may have problems as indicated above, and should consider changing back to the Release 16 ISMDO9 module.

OS/360 RELEASE 15/16 LINKAGE EDITOR F-V2

Block Editor	5-1	10-1	40-1
F44 F88 F128	All Values Acceptable	52K, 6K 88K, 36K 128K, 56K	80K, 6K 88K, 8K 128K, 28K

Exhibit 1

DOS/360 QTAM AUTOPOLL AND CPU UTILIZATION

The following item is from SECOM (360N-CQ-470-028).

The Autopoll feature considerably reduces the amount of CPU time used in polling leased lines. However, the CPU does not necessarily go into the wait state turning off the CPU meter. Approximately 500 ms are needed to go into the wait state. If the users polling list is sufficiently long, and he is negative polling, he will go into the wait state when QTAM is just polling. At the end of the poll list he will go out of the wait state to restart the polling list. Repeating polling entries and thus, creating a long polling list will save meter time when one is doing only negative polling.

DOS/360 PL/I 2314 USAGE

The following field contribution has not been submitted to any formal IBM test. Its usefulness should be evaluated in the users own environment prior to implementation.

Full system support for the 2314 under DOS PL/1 will be available later this year. For DOS Release 16 2314 installations, the following ways are available to utilize PL/1 on 2314 volumes now:

1. PL/1 work files and user data sets can be assigned to 2314 volumes which have been initialized as 2311's. At IPL time the 2314 volumes must be DELETED and ADDed as 2311's.

2. PL/1 work files (SYS001, SYS002, and SYSLNK can be run on 2314 drives initialized as 2314 drives by making minor changes to the compiler program. Essentially SYS001, SYS002, and SYS003 are defined within the compiler by DTFSD's which require the device type "2314". By changing these DTF's in modules IJXAØØ and IJXAØØD, the work files will be handled properly on the 2314. No change is required for SYSLNK since it is handled by the device independent DTFCP/CPMOD.

BPS/360 UTILITY INITIALIZE DISK ALTERNATE TRACK CORRECTION

The BPS Utility "16K Initialize Disk" (360P-UT-206) does not properly recognize flagged tracks in the alternate track cylinder (200 to 202). Additional information is available through FE RETAIN pertaining to an available "PTF" by referring to 360P-UT-206 with information key 011.

S/360 2916 LONG LINE ADAPTER RPQ

An RPQ unit is available which provides the capability for extending the channel interface up to two miles. A brief descrption is noted below. Additional information is available to you by contacting Custom Systems (Department 560), Kingston, New York. Supporting documentation, including a sales brochure is available on a limited distribution.

The IBM 2916 Long Line Adapter (LLA) provides a means of increasing the effective length of a System/360 selector channel interface up to two miles to accommodate remote control units and their associated devices, or to facilitate inter-system communications via a channel-to-channel adapter. Burst mode type devices having data rates up to 125,000 bytes per second may be connected via the LLA.

The LLA consists physically and logically of two main components, the 2916-3 and the 2916-4. Each performs logical functions which control transmission of data characters between the selector channel and the remotely located control unit(s). Standard interface bus lines are used for transmission of data characters and control signals between the selector channel and the 2916-3 and between the 2916-4 and the remote control units.

The 2916 LLA may be attached to System/360 Models 50, 65, 67 and 75 selector channels, and is attached to a selector channel in the same manner as any standard control unit.

The data rate between the local and remote LLA units is asynchronous to the channel and/or remote control unit(s). That is, while a fully duplexed bit stream of 125,000 bytes per second is flowing between the 2916 units, actual data is passed on an "as required" basis. This data

flow is controlled by a buffer in the remote 2916-4.

The LLA is provided with Extensive Diagnostic Programs, CD Panels, and maintenance plugboard to assist the Customer Engineer to maintain proper operation.

SUBMITTING TYPE III PROGRAM ABSTRACTS

Initial abstracts for programs which are to be submitted to the Type III Library should be forwarded to the Application Programming Standards Department (APSD) DPD HQ. Previously, these abstracts were mailed to TIE and were published in the TIE Catalog. Abstracts of Type III programs in development are no longer listed in the TIE Catalog - these abstracts are now broadcast daily via SECOM.

The preferred time to submit an Initial Abstract is before development begins; however, they may be submitted any time prior to forwarding the program material to the Program Information Department (PID).

The value of an Initial Abstract before the program is submitted is substantial because:

1. The field is apprised through SECOM that the program is in development. This serves to reduce duplicate effort and may place you in contact with others interested in the same endeavor. In many instances collaborated effort has resulted.

2. The review performed by DPD HQ on the Initial Abstract can save considerable effort if the program is judged ineligible for the Type III Library. Also, the review, in many instances brings forth suggestions which will enhance the program, or prevent delay in processing of the program when it is received by PID.

In submitting an Initial Abstract, follow the general guidelines below:

1. Use the Program Contribution Form (120-1420) for the Initial Abstract submission.

2. Follow the instructions contained in the Type III Standards and Procedures for Initial Abstract submissions. The Type III Standards and Procedures may be found in the DP Sales Activity Section of the Branch Office Manual, or a copy may be obtained from the Application Programming Standards Department.

3. Send the completed Program Contribution Form to the Application Programming Standards Department DPD HQ.

Do not send completed program material to APSD. Program material should be shipped directly to PID.

Questions regarding Type III program and abstract submission should be addressed to Mr. J. F. Schutty, Application Programming Standards Department, 112 E. Post Road, White Plains, N. Y. Telephone (914) 949-1900; Extension 5520 or IBM Dial Network 8-374-5520.

INFORMATION ABOUT THE NEWSLETTER

EVERY SALESMAN AND SYSTEMS ENGINEER SHOULD RECEIVE ONE COPY OF THE NEWSLETTER. Each FE Branch Manager should receive five copies for distribution to Customer Engineers.

The IBM Installation Newsletter is distributed only to IBM locations and is not available to customers. The Newsletter is separated into two parts. The first part contains information which is intended only for IBM personnel and the pages are labeled For IBM Internal Use Only. THESE PAGES, OR REPRODUCTIONS OF THEM, ARE NOT TO BE GIVEN TO CUSTOMERS. However, using your own good judgement, you may discuss pertinent information from these pages with a customer. The second part contains information which may be REPRODUCED by the Branch Office at their discretion and given to customers. These pages are not labeled with any restrictive classification. Note that when any abstracts for Type III programs are given to the customer, he must be informed that the disclaimer for Type III programs as contained in the Newsletter, applies. YOUR COOPERATION IN PROPERLY HANDLING THE CONTENT OF THE IBM INSTALLATION NEWSLETTER IS NECESSARY FOR ITS CONTINUED PUBLICATION.

Changes to the quantity of the Newsletter shipped are to be made by the Administration Manager. (NOTE THAT EXTRA COPIES OF THE NEWSLETTER ARE NOT TO BE ORDERED FOR DISTRIBUTION OF INFORMATION TO CUSTOMERS.) He should update and send the DP Administration Managers' Communication Control card (form number N44-780) to DP Document Control, DPD HQ.

Direct inquiries concerning the contents of this Newsletter to:

W. P. Gaffigan, Editor
IBM Installation Newsletter
Installation Planning-#713
DPD HQ
112 East Post Road
White Plains, New York 10601
Telephone 914 WH9-1900 Extension 4781
Tie Line 8-374-4781

NEW PUBLICATIONS ANNOUNCED BY PRL'S

The weekly PRL's (Publications Release Letters) are used to insure that all Salesmen and Systems Engineers are aware of new or revised Marketing Publications. Normally, each issue of the Newsletter will contain information from two PRL's, one following the other. The information will be placed in the Newsletter in its original form with no rearrangement of form numbers or titles. It is not intended to replace existing information and distribution sources. You should be certain that you are aware of these sources.

NOTE: Marketing Publications are obtained from The IBM Distribution Center Mechanicsburg, Pa.

The Publications Requisition, form number M02-0618-2, is used for ordering.

PRL #26 June 28, 1968

MARKETING PUBLICATIONS

FORM NO.	TITLE	DISPOSITION
N20-0361.30	IBM S/360 Mod 20 Re: A26-3565-4	Scrap N20-0361.29
221-0386-3	The 2701 Data Adapter	Scrap 221-0386-2
221-0399-2	The IBM 2020 Processing Unit	Scrap 221-0399-1
221-0428-1	The IBM 2703 Transmission Control	Scrap 221-0428-0
221-0441-2	IBM 2311 Disk Storage Drive, Models 11 and 12	Scrap 221-0441-1
221-0498-0	IBM System/360 Attached Support Processor (ASP)	NEW
270-0099-0	9950/54/65 Installation Instruction	NEW
320-0948-0	Proceedings of the 8th IBM Medical Symposium 1967	NEW
520-1300-1	Educating for Achievement	Use 520-1300-0
549-0003-1	Selectric-Composer Spec. Folder	Scrap 549-0003-0
549-0202-2	Composer Type Style Booklet	Use 549-0202-1
A26-3684-2	IBM 1080 Physical Planning	Scrap A26-3684-1 & N26-0187
A26-5893-6	IBM 2560 Multi-Function Card Machine Component Description and Operating Procedures	Scrap A26-5893-5
C20-1693-0	Functional Wiring Principles for IBM Punched Card Equipment	Use A24-1007-1
C24-5070-0	IBM S/360 BPS/BOS/TOS/DOS Tape Labels	NEW
E20-0306-1	IBM S/360 1287 Input Conversion Program (360A-DR-07X) Application Description Manual	Scrap E20-0306-0
H20-0475-1	IBM S/360 Automated Chemistry Program (360 ACP) for the 1080 Data Acquisition System Appl. Desc.	Scrap H20-0475-0
H20-0487-1	IBM S/360 Requirements Planning Application Description	Scrap H20-0487-0
H20-0514-0	IBM S/360 Model 44 Remote Access Computing System Application Description	NEW
H20-0515-0	IBM S/360 Automated Chemistry Program (360ACP) for the 1080 Data Acquisition System (360A-UH-12X) Program Description Manual	NEW
H20-0516-0	IBM S/360 Automated Chemistry Program (360ACP) for the 1080 Data Acuqisition System (360A-UH-12X) Operations Manual	NEW
H20-0528-0	IBM S/360 1287 Input Conversion Program (360A-DR-07X) Operations Manual	NEW
H20-0532-0	IBM S/360 Power System Planning for the Electric Utility Industry Application Description Manual	NEW
K20-0263-0	Mine Planning with the IBM 1130 System at Newmont Mining Co.	NEW
N20-1864	TNL for System/360 Bill of Material Processor (360A-ME-06X) Application Description with N20-1020 Re: H20-0197-2	NEW
N20-1865	TNL for System/360 Bill of Material Processor (360A-ME-06X) Operator's Manual with N20-1052 Re: H20-0254-1	NEW
N20-1870	TNL for System/360 Bill of Material Processor (360A-ME-06X) Version 2 - Programmer's Manual Re: H20-0246-1, - 2 with N20-1053 to -1 only	NEW
N21-0092	2560 SRL TNL to A26-5893-6	NEW
N21-5090	DOS/TOS RPG Spec. TNL Re: C26-3570-4, -5 with N24-5195, N21-5055, N21-5077 & N21-5086	NEW
N22-0297	Change to IBM S/360 Model 40 Functional Characteristics Re: A22-6881-2	NEW
N24-5360	TNL to TOS Operating Guide Re: C24-5021-3	NEW
N24-5361	TNL to IBM S/360 TOS Performance Estimates Re: C24-5020-4	NEW
N24-5362	TNL to IBM S/360 TOS System Generation & Maintenance Re: C24-5015-5	NEW
N33-1514	IBM S/360 Model 20 Introduction and System Summary Re: A26-5889-2 with N33-1514	NEW
N33-1515	IBM S/360 Model 20 Installation Manual - Physical Planning Re: A26-5896-5, -6 with N26-0164, N33-1500, N33-1503, N33-1511, N26-0154, N26-0180, N33-1502, N33-1508 & N33-1512	NEW
N33-1517	IBM S/360 Model 20 Configurator Re: A26-3572-5	NEW
N33-1518	IBM S/360 Model 20 Configurator Re: A26-3572-5 with N33-1517	NEW
R20-1055-2	Customer Education - Course Selection Guide	NEW - Reinstated
R20-1067-0	Fundamentals of Programming	NEW
R20-8042-0	IBM 1130/1800 FORTRAN IV Workshop Education Guide	NEW
R20-8058-0	IBM 1130 Computing System Continuous System Modeling Program (1130-CX-13X Type II Application Program) Education Guide	NEW
R20-8062-0	IBM Punched Card Data Processing, Accounts Receivable Education Guide	NEW
R20-9243-0	IBM 1130 Basic FORTRAN IV for Commercial Users	NEW
Y20-0175-0	IBM S/360 Automated Chemistry Program (360 ACP) for the 1080 Data Acquisition System (360A-UH-12X) System Manual	NEW

PRL #26 (Continued)

Y20-0210	TNL for System/360 Bill of Material Processor (360A-ME-06X) Version 2 - System Manual Re: Y20-0099-0	NEW
Y28-3080	IBM S/360 Time Sharing System Assembler PLM Re: Y28-2021-0 with Y28-3058 & Y28-3067	NEW
Y28-2016-1	IBM S/360 Time Sharing System Access Methods PLM	Scrap Y28-2016-0, Y28-3055 & Y28-3064
Y28-3077	IBM S/360 Time Sharing System System Service Routines PLM Re: Y28-2018-0 with Y28-3056 & Y28-3070	NEW
Z20-1866-1	IBM S/360 Model 25 Preliminary Marketing Guide	Scrap Z20-1866-0, Z20-1868 & Z20-1871
Z77-8061-0	Share XXIX Talk on Modeling Large Systems	NEW
Z77-8062-0	Luncheon Address Given at GPSS User Conference	NEW

PRL #27 July 5, 1968

N20-0360.48	IBM S/360 SRL Newsletter Re: A22-6822 -11	Scrap N20-0360.47
505-0010-0	A Computer Glossary	NEW
520-1966-0	The Computer Comes of Age	NEW
520-2079-0	IBM 1892 Model 10/Model 11 Operator Panels	NEW
520-2101-0	The IBM 2680 CRT Printer	NEW
520-2104-0	IBM S/360 Model 25 the new intermediate computing system with expanded capability	NEW-Use 520-2052-1
A24-3073-5	IBM 1403 Printer Component Description	Use A24-3073-4, N24-0398, N24-0403, N24-0410 & N24-0415
C24-3420-1	IBM S/360 BPS, BOS, DOS, TOS Programming Systems Summary	Scrap C24-3420-0
C24-9000-3	IBM S/360 Mod 20 Tape Programming System Control and Service Programs	Scrap C24-9000-2
C24-9003-3	IBM S/360 Mod 20 Tape Programming System Input/Output Control System	Scrap C24-9003-2
C26-5882-4	IBM 1800 Assembler Language	Scrap C26-5882-3 & N26-0512
H19-0007-0	IBM S/360 Model 20, Requirements Planning and Inventory Control System, Application Description Manual	NEW
N23-0620	1800 Data Acquisition and Control System (New Cover) Re: Z23-8004-0	NEW
N33-8012	IBM S/360 Operating System: ALGOL Programmer's Guide Re: C33-4000-0 with N33-8000 & N33-8002	NEW
N33-8535	IBM S/360 Mod 20 DPS/TPS Tape Sort/Merge Re: C26-3804-2	NEW
N33-8536	IBM S/360 Mode 20 DPS/TPS Tape Utility Programs Re: C26-3808-2 with N33-8528	NEW
N33-8538	IBM S/360 Model 20 UCS Utility Program Re: C26-3812-1, N33-8503 & N33-8515	NEW
R20-9259-0	DOS/TOS Facilities Non-Programmers Course Description	NEW - Scrap R20-9176
R29-0016-0	IBM Programming the 7070-7074 Handouts for Video Tape Lessons	NEW
V20-0169-0	IBM Plant Automation - Production Monitoring (slides)	NEW
V20-0174-0	IBM Operating System/360 MFT-II (slides)	NEW
V20-1077-0	Introduction to Plant Automation Systems	NEW
V20-0183-0	QTAM for Teleprocessing on IBM/360	NEW
Y20-0178-0	Advanced Life Information System (360A-IL-09X) Input Edit Program - System Manual	NEW
Y20-0183-0	Advanced Life Information System (360A-IL-09X) File Maintenance Include (R) Routines (Narratives)System Manual	NEW
Y20-0184-0	Advanced Life Information System (360A-IL-09X) File Maintenance Include (R) Routines (Flowcharts) System Manual	NEW
Y20-0185-0	Advanced Life Information System (360A-IL-09X) File Maintenance Call (S) Routines (Narratives) System Manual	NEW
Y20-0186-0	Advanced Life Information System (360A-IL-09X) File Maintenance Call (S) Routines (Flowcharts) System Manual	NEW
Y20-0187-0	ALIS (360A-IL-09X) File Maintenance Issues, Additions, Complex Changes, and Terminations Transactions - System Manual	NEW
Y20-0188-0	ALIS (360A-IL-09X) File Maintenance Simple Changes Transactions System Manual	NEW
Y20-0189-0	ALIS (360A-IL-09X) File Maintenance 36XX Financial Transactions - Payments System Manual	NEW
Y20-0197-0	Advanced Life Information System (360A-IL-09X) Output Analysis Program - System Manual	NEW
Y20-0198-0	Advanced Life Information System (360A-IL-09X) Transaction Register Program - System Manual	NEW
Y20-0199-0	Advanced Life Information System (360A-IL-09X) Status Print Program - System Manual	NEW

PRL #27 (Continued)

Y20-0200-0	Advanced Life Information System (360A-IL-09X) Policy Accounting Journal - System Manual	NEW
Y20-0201-0	Advanced Life Information System (360A-IL-09X) Accounting Control Program - System Manual	NEW
Y20-0202-0	Advanced Life Information System (360A-IL-09X) Error Register Program - System Manual	NEW
Y20-0203-0	Advanced Life Information System (360A-IL-09X) Rate File Program - System Manual	NEW
Y28-3081-0	IBM S/360 Time Sharing System Resident Supervisor PLM Re: Y28-2012-0 with Y28-3052 & Y28-3062	NEW
Z77-8052-0	A Clinical Laboratory Information System Utilizing Card Model 20/1080 Systems	NEW
Z77-8063-0	Installation of IBM 1001 Teleprocessing Network in Hospital Patient Charge Coll.	NEW
Z77-8064-0	Teleprocessing Study and Design	NEW
Z77-8065-0	DOS BTAM Coding for Leased Line IBM 2780 Data Transmission Terminal	NEW
Z77-8066-0	General Guidelines for Installing a Standard CCAP Package	NEW

July 26, 1968

Issue No. 68-14

IBM INSTALLATION NEWSLETTER

TABLE OF CONTENTS

OS/360 GIS DATA MANAGEMENT CLARIFICATION

GIS (Generalized Information System) uses OS/360 standard Data Management facilities. For user file processing, these facilities include, BSAM, QSAM, BISAM, and QISAM. Therefore, any file created by GIS can be processed using programs based on any other OS/360 language so long as the other language can properly access the internal structure of the file. This represents a rather trivial inter-dependence in most standard data processing applications. There are, however, some possible pitfalls. It is possible to create a GIS file which is hierarchically structured with nested sets of repeating data. Not all of the multiple levels of "occurs on" dependencies. GIS files are, however, designed and created by the user using the GIS Data Description Language. The user, not GIS, determines the hierarchical structure of GIS files. It is his responsibility to determine whether or not his file organization can be processed by another OS/360 language.

If files are created by OS/360 languages other than GIS, and if the files can be described to GIS via the Data Description Language, then GIS can process and/or maintain these files. If the other languages, in creating the files, inserted slack bytes or if the other languages required slack bytes in processing the files created by GIS, then the user is totally responsible for describing the presence of or the requirement for these slack bytes to GIS via the DDT. These slack bytes are required for boundary alignment in other GIS than languages. A specific example is inappropriate since it would be dependent on the file organization required by the other language and the application being processed.

DOS/360 RELEASE 16 QTAM PUTTING 1033 IN TEXT MODE

The user must supply circle D in front of his printable data to put the 1033 printer in text mode to print. He must also insert 3 blank or unprintable characters between each printable character. This can be done by using Pause statements, or by insertions in the processing program.

DOS/360 COBOL RECOMMENDATIONS

The following recommendations are intended to provide a concise reference to assist in achieving efficient DOS/360 COBOL compile and object run times:

INPUT-OUTPUT

1. Open all files with one open statement and close all files with one close statement if possible.
2. Have only one read instruction per Input file and only one write instruction per Output file; e.g., 10 write after advancing statements, will generate 280 instructions.
3. Make recording F or U records a multiple of 8 bytes. Make recording V records an odd multiple of 4 bytes.
4. Write after advancing cannot be used on blocked utility files.
5. Initializing a record in the file section, e.g., move spaces, will not initialize subsequent records in the buffers.
6. "Read into" and write from" - see MOVE.

NUMERIC FIELDS

7. Zoned decimal fields on which no arithmetic will be done, should be picture "X" even when they contain only numerics.
8. Fields on which arithmetic will be done, should be Packed or Binary whenever possible, i.e., except unit record files. Therefore, convert fields of present data files requiring arithmetic to Packed or Binary.
9. Avoid mixed modes when possible. Move arithmetic display fields to working storage fields defined as Packed or Binary to avoid multiple conversion.
10. Study the decimal requirements of your present files, and then align the decimals of related fields on the converted data file.

NUMERIC LITERALS

11. Write numeric literals with the same number of decimal positions as the receiving field.

12. Use a sign on numeric literals unless a hexidecimal F is required. The "S" on the receiving picture in a move statement, will not produce a hexidecimal C.

PACKED DECIMAL

13. In general, most commercial applications are safer with numeric fields in Packed format.

14. Fields which have decimal positions should be packed.

15. Fields that require print editing will save core if they are defined as Packed.

16. Use Packed fields when the number of digits exceeds 9 positions.

17. If Packed is used, the picture should begin with "S" and the field length should be an odd number.

BINARY

18. Subscripts and "depending on" field of to go depending on and "occurs depending on" should be Binary.

19. If Binary is used, the picture must begin with "S" and the field length should be 4, 9 or 18 with no decimal positions.

FLOATING POINT

20. Avoid floating point except for purely scientific applications.

EDITING

21. Do not ask for total zero suppression if not required.

22. Floating edit symbols better than fixed.

23. If the last character of a picture is a period or comma, a period or comma must follow.

SUBSCRIPTING

24. Subscripts with literals when possible. Resolved at compile time.

25. Minimize subscripting.

ZONE STRIPPING

26. To strip zones except lower order:

    Move from Unpacked to Packed
    Move back Packed to Unpacked

CONDITIONAL BRANCHING

27. Different size fields require extra core in IF statements.

28. "Go to depending on" is better than several IF statements. If the code values are irregularly spaced in the collating sequence, a Binary Table search of a code table to obtain the value for the "go to depending on" parameter is still better than many IF statements.

MOVE

29. Move short literal i. e., less than 5 characters, better than move figurative constant or move all.

30. Different size fields require extra core in move statements.

31. Avoid move more than 256 bytes at a time unless a very large area is involved.

32. Avoid "read into" and "write from" if record exceeds 256 bytes unless very large record.

33. Move group, including "read into" and "write from", containing "occurs depending on" is controlled by the "depending on" parameter of the receiving field.

34. Move of a Packed field to an"X" field will not Unpack.

35. Move of a Binary field to an "X" field will not convert and Unpack.

PERFORM AND ALTER

36. If possible, use alter and go to statements, rather than perform, when minimum code is required.

37. An alter of coding other than a "go to" will not diagnose at compile time.

GENERAL

38. Keep statements simple.

39. If core is very restricted, remove all coding which causes include statements in the link edit listing, e.g., accept display exhibit stop literal.

40. If still over core, use overlays which are very easily set up - see C24-5025.

DOS/360 COBOL DYNAMIC CORE DUMP SUBROUTINE

The following is a field contribution which has not been submitted to any formal IBM test. Potential users should evaluate its usefulness in their own environment prior to implementation.

It is sometimes useful to know the status of certain core areas during the debugging stage of a program. "SNAPSHOT" is a subroutine which can be link-edited with a COBOL program to provide a partial core dump during main program execution. The main COBOL program calls the subroutine which, in turn, utilizes the PDUMP Assembler macro to dump core between any limits desired. Control is then returned to the COBOL program. SNAPSHOT, shown in Exhibit 1, may be called at any point in the COBOL program, and the limits of the dump can be altered under program control. Other points to note are:

1. SNAPSHOT uses General Registers 0-3 and so the contents of these registers as printed will differ from their contents during COBOL execution.

2. If SNAPSHOT is catalogued to the Relocatable Library, the Autolink feature of the Linkage Editor will include the SNAPSHOT module without the need for an INCLUDE statement.

Data Division working storage entries for the dump limits can be made as follows:

```
01 LIMITS        USAGE IS COMPUTATIONAL.
   02 LOWER                PICTURE S99999.
   02 UPPER                PICTURE S99999.
```

The use of the 01 level group entry assures full-word boundary alignment for the values of LOWER AND UPPER. The USAGE must be COMPUTATIONAL and the VALUE clause may be used if fixed limits are desired. Note that the limits must be specified in decimal rather than hexadecimal, i.e., a dump between core locations X'2000' and X'200C' would require that LOWER be assigned a value of 8000 and UPPER 8012.

Procedure Division entries needed to call SNAPSHOT are:

```
ENTER LINKAGE.
CALL 'SNAPSHOT' USING LOWER, UPPER.
ENTER COBOL.
```

1130 COMMERCIAL SUBROUTINE PACKAGE-PROGRAMMERS REFERENCE CARD

The new SRL manual for Version 3 of CSP contains an extremely useful item which should not be overlooked -- a programmers reference "card". However, since it is not mentioned anywhere in the body of the manual, it may go unnoticed.

This "card", printed on both sides of a standard 8 1/2 X 11 manual page, contains the calling sequences for all the CSP routines, with comments on their use; the decimal equivalents of the various EBCDIC characters; SKIP codes for the 1132 and 1403; and several other often-used tables of parameters.

Each copy of H20-0241-3, the CSP Program Reference Manual, contains <u>two</u> copies of the reference "card" -- one is on pages 195 and 196, the other (an unnumbered page) immediately following it. The second one is designed to be removed from the manual.

CORRECTIONS TO SRL FORM NUMBERS

Installation Newsletter 68-13 contained two articles referencing obsoleted form numbers.

1. "DOS/360 PL/I Coding Efficiency Reference List": (C28-6594-2 has been obsoleted by (C28-6594-3).

2. "S/360-20 DPS Use of DTFEN Ovlay": (C24-9007-3) has two TNLs to it, TNL-N33-9037, TNL N33-9030.

DOS/360 COBOL DYNAMIC CORE DUMP SUBROUTINE

```
STMT   SOURCE STATEMENT                                          DOS CL2-5 04/21/68

    1 SNAPSHOT START 0                                                        SNAP005
    2 BEGIN    BALR  15,0                                                     SNAP010
    3          USING *,15                                                     SNAP015
    4          ENTRY SNAPSHOT      ENTRY POINT FOR CALLING PROGRAM            SNAP020
    5          SAVE  (14,12)       SAVE CONTENTS OF COBOL REGISTERS           SNAP025
    6** 360N-CL-453  SAVE      CHANGE LEVEL  2-0
    7+         STM   14,12,12+4*(14+2-(14+2)/16*16)(13)
    8          LM    2,3,0(1).     LOAD ADDRESSES OF LIMITS INTO R2 AND R3    SNAP030
    9          L     2,0(2)        LOAD LOWER LIMIT INTO R2                   SNAP035
   10          L     3,0(3)        LOAD UPPER LIMIT INTO R3                   SNAP040
   11          PDUMP (2),(3)       DUMP CORE BETWEEN UPPER AND LOWER LIMITS   SNAP045
   12** 360N-CL-453  PDUMP     CHANGE LEVEL  2-0
   13+         CNOP  0,4
   14+         ST    2,*+16
   15+         ST    3,*+16
   16+         LA    1,=CL8'$$BPDUMP'
   17+         BAL   0,*+12
   18+         DC    A(2) ADDRESS 1
   19+         DC    A(3) ADDRESS 2
   20+         SVC   2
   21          RETURN (14,12)      RESTORE CONTENTS OF COBOL REGISTERS        SNAP050
   22** 360N-CL-453  RETURN    CHANGE LEVEL  2-0
   23+         LM    14,12,12+4*(14+2-(14+2)/16*16)(13)
   24+         BR    14
   25          END   BEGIN                                                     SNAP055
   26                =CL8'$$BPDUMP'
```

DOS/360 COBOL DYNAMIC CORE DUMP SUBROUTINE

```
LINE NO.  SEQ. NO.        SOURCE STATEMENT              CBD CL2-5 04/21/68

      1          IDENTIFICATION DIVISION.
      2          PROGRAM-ID. 'TESTSNAP'.
      3          ENVIRONMENT DIVISION.
      4          CONFIGURATION SECTION.
      5          SOURCE-COMPUTER. IBM-360 E30.
      6          OBJECT-COMPUTER. IBM-360 E30.
      7          DATA DIVISION.
      8          WORKING-STORAGE SECTION.
      9          01  LIMITS                        USAGE IS COMPUTATIONAL.
     10              02  LOWER                     PICTURE S99999 VALUE 8192.
     11              02  UPPER                     PICTURE S99999 VALUE 9768.
     12          01  TESTLIT      PICTURE X(10) VALUE 'TOM STRONG'.
     13          PROCEDURE DIVISION.
     14          START.
     15              ENTER LINKAGE.
     16              CALL 'SNAPSHOT' USING LOWER, UPPER.
     17              ENTER COBOL.
     18              STOP RUN.
```

Exhibit 1

NEW TYPE III PROGRAMS

The following are the abstracts of programs which have been recently made available from the Type III library.

The program, along with its complete abstract, will be incorporated in subsequent issues of the Catalog of Programs.

Programs may be obtained by submitting a properly completed "General Program Request Card" (Form Number 120-1145-1) to the Program Information Department, 40 Saw Mill River Road, Hawthorne, New York, 10532.

These programs and their related documentation are distributed by IBM in the author's original form and have not been subjected to any formal testing.

Any discussions of Type III programs with customers must emphasize the following points:

1. Type III (IBM employee-contributed) programs are provided by the IBM Corporation as part of its service to customers.

2. Type III programs have not been subjected to any formal test.

3. Recipients of Type III programs are expected to make the final evaluation as to the usefulness of the programs in their own environment.

4. There is no committed maintenance for Type III programs. However, any changes the author chooses to make will be announced in subsequent issues of the Catalog of Programs.

5. IBM makes no warranty, expressed or implied, as to the documentation, function, or performance of Type III programs.

-----REVISION-----

A PROGRAM FOR EMPTY FREIGHT CAR ALLOCATION.

Program deck has been resequenced. No other changes.

Ordering Procedure: Order File Number 360D-29.4.002.

The program material can be obtained in card form or on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR). If not specified, a 9 track DTR will be provided. No tape submittal is required. The DTR will be provided by the Library.

1401 SIMULATOR FOR OS/360. This program allows the execution of 1401 programs on a SYSTEM/360 Model 30-75, under control of the Operating System, PCP, MFT, or MVT. No special hardware is required. 1401 features supported are advanced programming, sense switches, tapes, multiply, divide, 16K core, and all standard instructions except Select Stacker. Operator control is through the 1052 and includes tape and sense switch assignment, load from cards or tape, display and alter core, start reset, start, clear core, write tape mark, and rewind. The program is written in OS/360 Assembler Language. The minimum requirements are the same as for OS/360 and must

include I/O devices used for simulation. If tape simulation is desired, tape drives must also be available. The simulator program requires approximately 70,000 bytes of core.

Ordering Procedure: Order File Number 360D-11.1.019.

The program material can be obtained on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR). If not specified, a 9 track DTR will be provided. No tape submittal is required. The DTR will be provided by the Library.

DOS PROOF OF DEPOSIT TRANSIT. This set of programs is an expanded version of the S/360 Proof of Deposit Transit programs number 360D-19.7.002. It will operate in a multiprogramming environment with one reader/sorter, one printer and one disk per partition (maximum of 3 partitions under current version of DOS). It features: multibank capabilitity with float analysis by account and bank, full page approach, non-stop reader/sorter processing with divider slips for triggering bundle totals, Pass to Pass control, Destination Headers for efficient sorting of high number of endpoints, DDA transaction capturing selective by bank with deposit analysis, automatic cash letters and selective multiple copy recaps, endpoint analysis, sort pattern generator and all file create programs needed for operation.

Minimum configuration is a S/360 32K Mod 25 decimal feature, one selector channel, 1419, 1403, 2-2311's, 2540 (2450 not used during transit operation) and 1052 console. The source language is Basic Assembler under DOS.

Ordering Procedure: Order File Number 360D-19.7.008.

The program material can be obtained on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR). If not specified, a 9 track DTR will be provided. No tape submittal is required. The DTR will be provided by the Library.

1092/93 KEYMAT DECODING WITH MEDICAL INFORMATION SYSTEM PROGRAMS. This program provides a means for decoding a keymat for 1092/93 programmed keyboard for a MISP (Medical Information System Programs) installation. The 1092/93 Keymat Decoding Table Load Utility, DEPZMUDC, loads a formatted 2311 data set from cards containing information associated with a specific button for a given keymat. The macro, MISLDCAN, enables a user to obtain this interpretative information using column and button numbers as search parameters.

Both the Utility and the Macro are written in System/360 DOS Assembler Language.

The programs may also be used with TESTMISP (360D-04.4.007).

Ordering Procedure: Order File Number 360D-06.4.002.

The program material can be obtained in card form or on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR). If not specified, a 9 track DTR will be provided. No tape submittal is required. The DTR will be provided by the Library.

DOS/MICR SIMULATOR. This program is a package of macro definitions and a relocatable module which will simulate for the user's program as many as three MICR reader/sorters. Functions normally performed by the DTFMR, DTFMR I/O macros, and DTFMR logic module are assumed by the simulator. Instead the user incorporates into his program the simulator's I/O Module and special macros which are functionally identical to those provided by the DOS system. Consequently, when the user no longer needs simulation services, his MICR program can be reassembled with the standard macros for software support. Simulated MICR input to the program is from a card reader and/or as many as two magnetic tape units.

The simulator and its macro definition are coded in DOS/360 Assembler Language. Object machine requirements are the same as those for the DOS System to be simulated with the exception that a card reader or tape unit replaces each 1419 to be simulated. It is not required that the user's DOS Supervisor have the MICR

option.

Ordering Procedure: Order File Number 360D-11.2.002.

The program material can be obtained in card form or on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR). If not specified, a 9 track DTR will be provided. No tape submittal is required. The DTR will be provided by the Library.

CONVERTING A NUMBER TO ITS WORD EQUIVALENT ROUTINE (CANTIWE). This is a routine that allows the user to "write out" any number up to seven digits (including two decimal places) in its equivalent word representation. For example, the amount $1125.32 would be printed, "ONE THOUSAND ONE HUNDRED TWENTY FIVE AND 32/100 DOLLARS. This routine can be very useful when printing Payroll Checks, Accounts Payable Checks, or any commercial application that involves amounts to be printed.

The routine is written in RPG and can be inserted into any RPG program for any S/360 operating system with minimum core capacity of 8K.

Ordering Procedure: Order File Number 360D-06.5.004.

The program material can be obtained in card form or on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR). If not specified, a 9 track DTR will be provided. No tape submittal is required. The DTR will be provided by the Library.

TOTAL OPERATOR ORIENTED BICAR APPLICATION. This 1130 package consists of generalized programs for billing, inventory control, accounts receivable and sales analysis using the 4K-1130 Computing System. The application is designed to be operator oriented using the 1130 paper tape-disk system. The creation of invoices, the updating of inventory receipts, and the applying of cash to the accounts receivable is accomplished by the operator entering the necessary information from the console keyboard. As a result of these operations all records are updated and all other reports can be automatically produced on the console printer. Some of the reports included in in this package are a below minimum, stock status, sales analysis by item, by customer, by salesman, and by product class, aged trial balance and aged statements. The system is designed to utilize the console switches to tailor the system to a particular customer. All major programs are written in FORTRAN for easy modification. Estimated volumes are 2,000 customers, 6,400 inventory items, 750 invoice lines/day, and 8,000 open accounts receivable records.

Ordering Procedure: Order File Number 1130-28.2.001.

Distribution will be in card form only.

IBM Installation Newsletter

August 23, 1968 Issue No. 68-16

CONTENTS
(G means Green Section)

Distribution: Branch Offices - DP Management, Salesmen, Systems Engineers, FE Managers. Regions, Districts, Education Centers, Field Systems Centers, Federal Systems Centers, FE Area Offices, DPD HQ, FED HQ, WTC.

(Table of Contents Continued)

*Requires Immediate Attention

For IBM Internal Use Only

DPD HQ, White Plains, N. Y.

SELECTED SHORT SUBJECTS

The purpose of Selected Short Subjects is to bring together and highlight concise, factual and timely information which will indicate that action is to be taken by the IBM representative whose accounts are affected.

1. OS/360 COBOL E USE OF OCCURS AND REDEFINES

There is an article in this issue of the Newsletter titled "DOS/360 COBOL Use of OCCURS and REDEFINES" which also applies to OS/360 COBOL E.

2. 2740/41 Print Quality Considerations

If you have a 2740/41 on order or installed you should read the article with the above title in this issue of the Newsletter.

OS/360 - PLCS TO REDUCE HARD WAIT STATES

Installation Newsletter 68-10 contained, on page 6, an article on the above subject. The following information presents the current status on PLC's to reduce hard wait states and is an extension of the article in 68-10. Issue 68-10 used the term PTF. This article uses the term PLC (Program Logic Change). The difference is that PTFs are correlated to APAR numbers while PLC's are not necessarily so correlated. Hence, PLCs can not always be associated with a particular APAR. However, PLCs are ordered by a "PTF" number.

There are now Program Logic Changes (PLC's) available to address the problem of system wait states in OS/360. The PLC's are designed to maximize software system availability through the implementation of a "fail soft" technique. The PLC's increase system availability by reducing the impact of system wait states, e.g., F03 wait states, on system operation. The increase in system availability is achieved by:

1. Isolating the ABENDing region (MVT) or partition (MFT) when a system wait state would have occurred, and

2. Allowing the other regions and partitions to continue processing.

The net result is a fail soft effect accompanied by the unavailability of certain system resources. The ABENDing region or partition remains in the same state as existed when the ABEND occurred, thereby permitting the problem to be analyzed when a stand-alone core dump can be taken. The computer operator is informed of the system's damaged state by a message on the console typewriter indicating the ABENDing task name or partition identification and the wait state code that would have occurred.

These PLC's have been tested in a number of customer environments and have proven to substantially reduce the number of re-IPL's, as well as allowing the user in most cases to re-IPL at his convenience.

Exhibit 1 contains a chart of applicability of the fail soft PLC's. The correct PLC can be determined from the chart. Use the number indicated as a PTF number and order by normal PTF procedures.

NOTE: PTF 12205 will bypass the current job if the ABEND occurs in the scheduler and refresh the scheduler to continue reading the input stream.

OS/360 PCP/MFT-II/MVT JOB STREAM CONSIDERATIONS - CORRECTION

Installation Newsletter 68-11 contained an article on the above subject. The following is a correction to item #11, page 9, Multiple OPEN's for Sysin or Sysout.

The information regarding handling of Sysout Data sets is incorrect and should be replaced by the following.

Sysout data sets are opened with an assumed disposition of MOD. Each time a Sysout data set is opened, writing will begin from the end of the last record written. All data written to a Sysout data set, regardless of the times closed and opened, will be processed by the Output Writer.

OS/360- PLCS TO REDUCE HARD WAIT STATES

"FAIL SOFT"

PTF APPLICABILITY

Option - OS/Release	EMFT	MFT I	MFT II	MVT
13	11943-13	11943-13, 12205-13	-	-
14	11943-13	11943-13, 12205-14	-	12721-14
15	-	-	11943-15	12721-14

Exhibit 1

OS/STRAM READ/WRITE MACRO RECOMMENDATION

OS/STRAM provides two levels of macro instructions for transmitting data to and from STR terminals -- the basic macro level (READ/WRITE) and the queued level (GET/PUT). The READ/WRITE level is recommended for new/rewritten programs for the following reasons:

1. Considerable less core storage (approximately 2.0 to 2.5K) is required for STRAM; this storage can be used by the problem program.

2. By using OS/360 Buffer Management at the READ/WRITE level, buffer storage requirements are reduced and user buffer-management programming requirements are simplified significantly. (For example, the queued level imposes a user requirement to retrieve data in all buffers that have been filled but not yet written when an I/O error occurs while writing a prior buffer.) At most installations, double buffering at the READ/WRITE level will provide line utilization equivalent to the GET/PUT level.

3. Less CPU time is required to process a message.

4. Maintenance problems are reduced because the user program does not have to operate in conjunction with the complex Ring Channel program required by the queued level.

5. Conversion to BTAM/BSC will be facilitated. This is an important factor since STR terminal equipment is being displaced by Binary Synchronous equipment.

DOS/360 MULTI-PROGRAMMING - REDUCING 1052 CONTENTION

The following is a field contribution which has not been submitted to any formal IBM test. Its usefulness in the user's own environment should be evaluated prior to implementation. No maintenance can be expected.

At times it is necessary for the problem program to request an operator action or decision before continuing to process. The console typewriter is an ideal means of exchanging the required information. This causes problems of contention for the console which may be critical in a multi-programming environment. Using the following technique, the console will not be tied up with a pending read while the operator performs the requested action or decision. The operator enters his answer upon completing his action and the console has been free in the mean time. The following numbered items correspond to the circled numbers in Exhibit 1.

1. At beginning of program use STXIT to set operator communications exit.

2. After output of desired message branch to operator wait routine.

3. Operator wait routine (OPWAIT) issues a wait which is not satisfied.

4. When operator interrupts control passes to operator communication routine (OPCOM). Routine requests operator to "Enter Response" previously asked for and executes a NOP to satisfy wait issued in (3). Control then returns to supervisor.

5. CCB with a NOP CCW for console (SYSLOG) and save area for supervisor useage.

DOS/360 QTAM AUTOPOLL AND CPU UTILIZATION

Newsletter issue 68-14 contained on page 5, an article on the above subject. The following replaces that article in its entirety.

Autopoll feature considerably reduces the amount of CPU time used in polling leased lines. However, the CPU does not go into the WAIT state turning off the CPU METER. If the users polling list is sufficiently long, and he is negative polling, he will go into the WAIT state when QTAM is just polling. At the end of the poll list he will go out of the WAIT state to restart the polling list. Repeating polling entries and thus, creating a long polling list will save CPU time when one is doing only negative polling.

IBM

IBM System/360 Assembler Coding Form

PROGRAM REDUCING 1052 CONTENTION | PUNCHING INSTRUCTIONS | GRAPHIC | PAGE 1

PROGRAMMER CVC | DATE 7/19/68 | PUNCH | CARD ELECTRC

STATEMENT

Name	Operation	Operand	Comments
SAMPLE	START	Ø	
	BALR	12, Ø	12 IS BASE REG
	USING	*, 12	
	STXIT	OC, OPCom, SAVE	SET OPERATOR COMMUNICATION (1)
	LA	1Ø, REELMESS	MESSAGE ADDRESS
	LA	9, L'REELMESS	LENGTH OF MESSAGE
	BAL	14, OUT1Ø52	OUTPUT MESSAGE TO 1052
	BAL	8, OPWAIT	BRANCH TO OPERATOR WAIT (2)
*			
* RETURN	TO THIS POINT WHEN OPERATOR IS		READY TO CONTINUE AND PROCESS
* ANSWER	READ IN OPCOM ROUTINE.		
	LA	1Ø, MESS2	MESSAGE 2
	LA	9, L'MESS2	LENGTH
	BAL	14, OUT1Ø52	OUTPUT ON 1052
	BAL	8, OPWAIT	TO OPERATOR WAIT (2)

Exhibit 1

IBM System/360 Assembler Coding Form

PROGRAM REDUCING 1052 CONTENTION | PUNCHING INSTRUCTIONS | GRAPHIC | PAGE 2
PROGRAMMER CVC | DATE 7/19/68 | PUNCH | CARD ELECTRO

```
*
** OPERATOR WAIT ROUTINE.
*
** THIS ROUTINE ISSUES A WAIT FOR CCB OPCTL WHICH IS NOT SATISFIED
** UNTIL OPERATOR ACTIVATES OPERATOR COMMUNICATION (OC) OPTION OF
** STXIT ISSUED AT BEGINNING OF PROGRAM.
*
OPWAIT   NI    OPCTL+2,X'7F'          SET OFF TRAFFIC BIT IN CCB   (3)
         WAIT  OPCTL
*
* CONTROL PASSES TO SUPERVISOR. PARTITION BECOMES INACTIVE UNTIL
* OPERATOR INTERRUPTS.
*
         BR    8                      RETURN TO PROCESS ANSWER     (6)
```

Exhibit 1 (Continued)

IBM System/360 Assembler Coding Form

PROGRAM REDUCING 1052 CONTENTION | PUNCHING INSTRUCTIONS | GRAPHIC | PUNCH | PAGE 3 | CARD ELECTRC

PROGRAMMER CVC | DATE 7/19/68

STATEMENT

Name	Operation	Operand	Comments
*			
** OPERATOR COMMUNICATION SUBROUTINE.			
*			
** CONTROL IS PASSED TO THIS ROUTINE FROM SUPERVISOR WHEN AN OPERATOR			
** RESTART IS INITIATED:			
*			
*	BG -	OPERATOR PRESS INTERRUPT KEY ON CONSOLE	
*	F2 -	MSG F2 COMMAND FROM ATTENTION ROUTINE (AR)	
*	F1 -	MSG F1 COMMAND FROM ATTENTION ROUTINE (AR)	
*			
OPCOM	LA	10,=C'ENTER RESPONSE'	ADDRESS
	LA	9,14	LENGTH
	BAL	14,OUT1052	
	LA	10,REPLY	ANSWER ADDRESS
	LA	9,3	ANSWER LENGTH
	XC	REPLY,REPLY	CLEAR ANSWER AREA
	BAL	8,IN1052	READ ANSWER
	EXCP	OPCTL	EXECUTE NOP
	EXIT	OC	RETURN TO SUPERVISOR

(4)

Exhibit I (Continued)

DOS/360 MULTI-PROGRAMMING-REDUCING 1052 CONTENTION

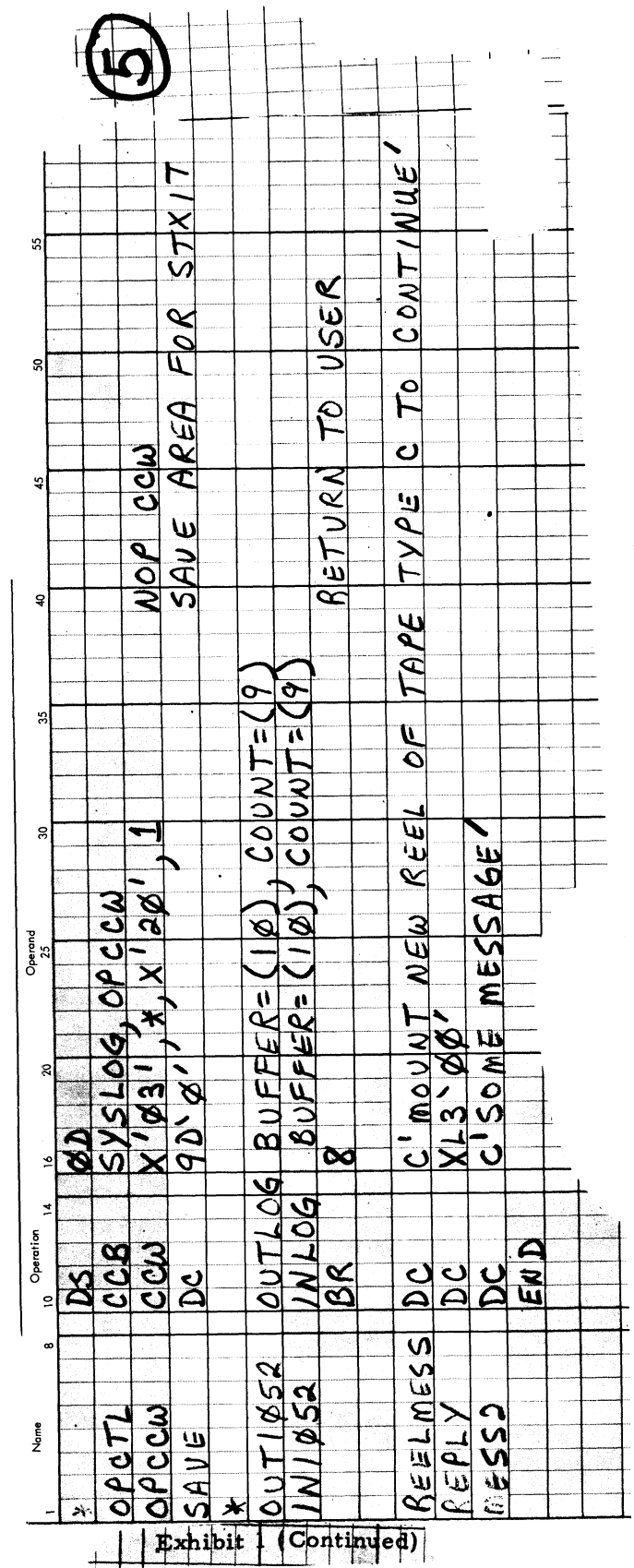


Exhibit I (Continued)

2740/41 PRINT QUALITY CONSIDERATIONS

Our 2740/41 customers are concerned not only with the entry, storage, retrieval and updating of information, but also with the quality of the final printout. Much of the success of the IBM Selectric® office typewriter is due to its ability to produce sharp, clear copy and many customers may expect the same results on the 2740/41.

It should be clearly understood, however, that the 2740/41 is not just like a typewriter. It was designed to be a communications terminal and certain features that assure high quality printing on a typewriter are not the same on the 2740/41. For example, it does not use the dual impression mechanism for impression control. This mechanism is used to provide the space function. Because of these mechanical differences, it is not realistic to expect Field Engineering to be able to raise the impression quality to the level designed into the IBM Selectric® office typewriter.

To obtain the best possible print quality on the 2740/41 the following should be considered:

1. Use a good quality, rag content paper, minimum #20.

2. Keep the element clean. Certain characters tend to fill up with dried ink and dust. This prevents sharp character definition.

3. Use a fresh ribbon. No typewriter can produce better quality than the quality of the ribbon and paper. Fabric ribbons tend to deteriorate more rapidly than others and should be changed as they do so. A darker ribbon than usual (no erasure smudge) and the IBM #60 Superior Write ribbon will provide darker print.

4. Photo-offset and multilith mat work quality are highly dependent upon uniform impression impact. This cannot be maintained on the 2740/41 at the same high quality as it is on the IBM Selectric® office typewriter. If multilith mats must be used, keep the following in mind:

   a. The customer must be cautioned not to expect the same quality available on IBM Selectric® office typewriters.

b. Do not have the impression control on 5. A light impression may suffice as long the impression is affixed to the mat (with a greasy ink, such as the All Purpose Nylon ribbon or Plate Writing ribbon). The higher setting aggravates the embossing of small, sharp characters. This can be counteracted to some extent by moving the multiple copy control lever one or two notches back.

5. Under no condition should Vellum or parchment paper be used.

6. Wax stencils also require extremely critical impression control and are not recommended, although the level of quality desired with stencils is usually not high. The same applies to Ditto masters.

7. A carbon ribbon is used to enhance print quality on the IBM Selectric® office typewriter. A carbon ribbon mechanism is not available on the 2740/41, pending availability of an RPQ which has been submitted. RPQ information is obtainable through your Regional Special Equipment Engineering Department.

IBM 2740 BASIC TERMINAL DESIGN CONSIDERATIONS

When considering the use of IBM 2740 basic terminals (with or without the Record Checking feature), be aware of the following limitations:

1. Under DOS and OS/BTAM support, there is a possibility of unsent messages if the application requires the sending of unsolicited messages to a terminal which has a READ INITIAL operation outstanding. A RESETPL is not valid for this operation due to a PREPARE command active to the channel. Therefore, a WRITE operation can only be issued if a READ is not pending.

2. SECOM entry 360S-cq-519-030 contains the following regarding QTAM support: 2740 basic or checking-only (2740 types 1 & 6) connected to a leased-line facility is supported in a manner similar to that provided for a switched line. Since the terminal is non-selectable by the CPU (via polling or addressing), output is deferred until the remote station has initiated transmission by sending a message to the CPU.

TELEPROCESSING PLANNING

The following outline is reprinted from a Chicago SD & I Center Newsletter. It was developed as a basic guide to the steps necessary for the design and implementation of a Teleprocessing system.

*Step One

1. Decide whether the application should be on-line and if so, if it must be "real-time".

   a. Is inquiry a part of the application?
   b. What is the response time requirement?
   c. Can the customer justify the on-line hardware cost?
   d. Is the programming staff adequate for the job?
   e. Must the on-line system process data in "real-time"? Can it collect input, batch process and distribute reports?

2. Select the right terminal or terminals.

   a. Determine volumes, both normal and peak.
   b. Plan for expansion - both normal expansion and that generated by the availability of on-line data.
   c. Determine I/O device requirements- keyboard, printer, card reader, etc.
   d. Consider the impact of terminal speed on communication line costs
   e. For data entry systems use the Data Entry Marketing Guide Z20-0841.

3. Decide on the way in which the CPU will be used.

   a. Dedicated to Teleprocessing for high volume applications with large processing requirements.
   b. Dedicated to Teleprocessing where program support requires it.
   c. Multiprogrammed for best use of the CPU.
   d. A two computer, front end/back end, system should receive the following considerations:

      1. Hardware cost.
      2. Program support for the front end/back end system.

3. Reliability requirements.
4. Back-up.

4. Plan early for backup hardware and procedures

a. Provide for use of nearby terminal as alternate, or provide other means of getting data to and from remote location.

b. Provide line backup if possible.

1. In dial systems backup is easy.
2. Leased lines can be backed up by use of dial facilities, but costs of data sets, switching equipment 2701/2/3 adapters and core storage for added program functions can be significant.
3. Cost of duplicate leased lines may eliminate this as a possibility.

c. Back-up 2701/2/3 for very high availability systems.

d. Back-up DASD devices as needed.

1. A second 2841 with 2311 switching will provide backup in a 2311 environment.
2. A 2844 provides effective 2314 control unit backup.
3. Switching 2311s or 2314s from an off-line system can be done with 8100 series features.

e. Plan to use a second CPU for backup if there is one in the installation.

1. If core on backup CPU is smaller, consider this in program design.
2. If CPU performance is less, plan for handling of priority transactions only.
3. Provide for switching of 2701/2/3 and DASD devices.

f. If a second CPU is not already installed, consider how to cost-justify a stand-by CPU.

1. Consider a terminal or terminals as CPU backup to send and receive priority input/output; or plan for moving input/output data in some other way.
2. Make arrangements with another user with a similar configuration for use of his system for batch processing of input/output.
3. Provide program facilities for card input to replace terminal input and printed output to replace terminal output.
4. Using a remote CPU for backup may be difficult because of:
   a. Lack of up-to-date master files.
   b. Added costs of lines and line switching equipment.
   c. Lack of experienced operators.
   d. Cost of 2701/2/3 stand-by at remote CPU.
5. Difficulties in updating local files to agree with transactions processed at remote back-up CPU.

5. Select an operating system. Although many non-Teleprocessing factors affect an operating system selection, some Teleprocessing considerations are:
   a. BPS or BOS for support of a minimum storage CPU as a remote terminal.
   b. DOS for low core storage requirements and ease of installation and operation.
   c. OS for added functions and facilities such as RJE, multitasking, DASD management, cataloged data sets, etc.

6. Select the right Teleprocessing programming support. Highlights of this support are:
   a. CCAP for minimum cost, single purpose teletype message switching systems.
   b. BTAM for low entry Teleprocessing systems and for support of high speed terminals.
   c. STRAM for support of STR terminals.
   d. QTAM for complex systems using low speed terminals.
   e. Industry oriented packages when applicable - SHAS, ALIS, ATS, etc.
   f. Type III programs where Type I or Type II support is not available.

7. Use the available design tools.
   a. CNDP for communication network design.
   b. TPAD for hardware configuration and for file organization and layout.

c. CSS for performance evaluation through modelling.
d. Performance evaluation literature such as:

1. Terminal timings in SRLs for 1050, 2740, 2780, etc.
2. BTAM timings in DOS performance
3. QTAM performance reports at SD & I Centers.
4. 2260 data entry performance data in Installation Newsletter. See the Newsletter KWIC Index in Issue 68-11 under 2260.

8. Select the best access methods.

a. Index sequential for ease of use and for combined random-sequential capability.
b. Direct access for fastest access to data.
c. Combined techniques:

1. Sequential with user created indices in core or on disk.
2. Index sequential tag file pointing to direct access master records.

d. Special, such as Bill of Material Processor organization.

9. Develop recovery procedures along with the system.

a. Build controls on transactions such as sequence numbers, batch totals and source terminal identification.
b. Provide transaction logging where required.
c. Provide for master file backup.
d. Use available checkpoint/restart facilities (such as, those provided by QTAM) if applicable.
e. Provide special restart facilities as required to avoid duplicate file updating or loss of transactions.
f. Provide messages to terminal operators to inform them of system status.
g. Develop and publish well defined operating procedures for recovery - both at terminals and at CPU.
h. Plan to use terminal test facilities of BTAM/QTAM.

10. Plan a step by step testing program

a. Testing of processing program logic with dummy transactions.
b. Simulating terminals with card input and printed output.
c. On site or datacenter testing with one or two locally installed terminals.
d. Datacenter use of TEST for thorough testing, peak volume checkout and throughput determination.

Steps One Through Ten

Keep it simple. Wherever possible, choose the straightforward over the complex technique of program design, file design, recovery, testing, etc.

REVISED TYPE III PROGRAM STANDARDS AND CONTRIBUTION FORM

Revised Type III program standards have been distributed in Branch Office Manual Revision #068-201, dated August 8, 1968.

A new Program Contribution Form is now available from the IBM Distribution Center, Mechanicsburg, Pa. The new form has an order number of 120-1424-3. The unit of measure is 01.

The changes in the Type III Standards fundamently deal with the use of the new Program Contribution Form; however, there are many other changes and clarifications to the standards and procedures for Type III program submissions.

Programs received at PID and Initial Program Abstracts received at APS after September 15, 1968 will be returned unless they use the new Program Contribution Form and follow the new Type III Standards.

Questions regarding the new material may be addressed to Mr. Jack Schutty, Application Programming Standards Department, DPD HQ White Plains, Phone (914) 949-1900 Extension 5520 or tie line 8-374-5520.

SUBMITTING NEW TYPE III PROGRAM ABSTRACTS CORRECTION

The article which appears under the above heading in issue number 68-14 of this newsletter erroneously indicated that the Type III Program Contribution Form was form number 120-1420. The correct order number for this form is 120-1424. Please make the necessary correction on page 6, the second paragraph from the bottom of the page, of Installation Newsletter 68-14.

INFORMATION ABOUT THE NEWSLETTER

EVERY SALESMAN AND SYSTEMS ENGINEER SHOULD RECEIVE ONE COPY OF THE NEWSLETTER. Each FE Branch Manager should receive five copies for distribution to Customer Engineers.

The IBM Installation Newsletter is distributed only to IBM locations and is not available to customers. The Newsletter is separated into two parts. The first part contains information which is intended only for IBM personnel and the pages are labeled For IBM Internal Use Only. THESE PAGES, OR REPRODUCTIONS OF THEM, ARE NOT TO BE GIVEN TO CUSTOMERS. However, using your own good judgement, you may discuss pertinent information from these pages with a customer. The second part contains information which may be REPRODUCED by the Branch Office at their discretion and given to customers. These pages are not labeled with any restrictive classification. Note that when any abstracts for Type III programs are given to the customer, he must be informed that the disclaimer for Type III programs as contained in the Newsletter, applies. YOUR COOPERATION IN PROPERLY HANDLING THE CONTENT OF THE IBM INSTALLATION NEWSLETTER IS NECESSARY FOR ITS CONTINUED PUBLICATION.

Changes to the quantity of the Newsletter shipped are to be made by the Administration Manager. (NOTE THAT EXTRA COPIES OF THE NEWSLETTER ARE NOT TO BE ORDERED FOR DISTRIBUTION OF INFORMATION TO CUSTOMERS.) He should update and send the DP Administration Managers' Communication Control card (form number N44-780) to DP Document Control, DPD HQ.

Direct inquiries concerning the contents of this Newsletter to:

W. P. Gaffigan, Editor
IBM Installation Newsletter
Installation Planning-#713
DPD HQ
112 East Post Road
White Plains, New York 10601
Telephone 914 WH9-1900 Extension 4781
Tie Line 8-374-4781

NEW PUBLICATIONS ANNOUNCED BY PRL'S

The weekly PRL's (Publications Release Letters) are used to insure that all Salesmen and Systems Engineers are aware of new or revised Marketing Publications. Normally, each issue of the Newsletter will contain information from two PRL's, one following the other. The information will be placed in the Newsletter in its original form with no rearrangement of form numbers or titles. It is not intended to replace existing information and distribution sources. You should be certain that you are aware of these sources.

NOTE: Marketing Publications are obtained from The IBM Distribution Center Mechanicsburg, Pa.

The Publications Requisition, form number M02-0618-2, is used for ordering.

PRL #30 July 26, 1968

REFERENCE SOURCES

FORM NO.	TITLE	DISPOSITION
N20-1853-0	TNL to BAP - Distribution Industries - Re: H20-0522-0	NEW

MARKETING PUBLICATIONS

270-0026-1	9951/56 Instruction Maintenance Manual	NEW
270-0100-0	9370 Service Index	NEW
504-7301-0	IBM's Relationship with Employment Agencies and Search Firms	NEW
520-1261-1	IBM S/360 for Scientists and Engineers	Scrap 520-1261-0
543-0066-0	Selectric Short Course Booklet	NEW
570-0414-0	Information Recorder Layout Form W/Numbers	NEW
570-0415-0	Information Recorder Layout Form	NEW
A22-6889-2	IBM S/360 Model 75 Functional Characteristics	Scrap A22-6889-1
A24-3255-7	IBM S/360 Model 30 1401/1440/1460 Compatibility Feature	Scrap A24-3255-6
C20-1690-0	IBM 1130 Computing System User's Guide	NEW
C26-3715-3	IBM 1130/1800 Basic FORTRAN IV Language	Scrap C26-3715-2
H20-0368-1	IBM S/360 Continuous System Modeling Program (360A-CX-16X) Operator's Manual	Scrap H20-0368-0
J20-0038-1	CALL/360: DATATEXT Operator Reference Manual	NEW
L22-6926-0	IBM Airlines Reservation System Custom Feature Description 1977 Terminal Unit Model 1-- RPQ 828027	NEW
N20-1096	TNL for Advanced Life Information System Policy Master Record Code Book Re: H20-0483-0	NEW
N21-0094	1231/1232 TNL Re: A21-9012-6	NEW
N22-0306	TNL - Change to C22-6820-8 (1287 Models 3 & 4, 1288)	NEW
N28-0244	IBM S/360 Disk and Tape Operating Systems: COBOL Language Specifications Re: C24-3433-4 to -3, N24-5188, N28-0232, N28-0237, to -3 & -4, N28-0241	NEW
N33-1519	IBM S/360 Model 20 Functional Characteristics Re: A26-5847-4	NEW
N33-1520	IBM S/360 Model 20 Installation Manual - Physical Planning Re: A26-5896-5, -6 with -5, N26-0154, N26-0164 & N26-0180 with -5 & -6, N33-1500, N33-1502, N33-1503, N33-1508, N33-1511, N33-1512 & N33-1515	NEW
R20-9011-3	1401 DPS Executive Course	Scrap R20-9011-2
R20-9079-2	407 Accounting Machine Operation and Wiring Programmed Instruction	Scrap R20-9079-1
R20-9083-2	IBM 519 Document - Originating Machine Operation and Wiring Programmed Instruction	Use R20-9083-1
R20-9154-1	IBM S/360 DOS Communications Coding -- BTAM	Use R20-9154-0
R20-9234-0	IBM S/360 Continuous System Modeling Program	NEW
R29-0152-3	402-3 Operation & Wiring Notebook	Scrap R29-0152-2
V20-0164-0	IBM 50 Magnetic Data Inscribed and IBM 2495 Tape Cartridge Reader (slides)	NEW
V20-0176-0	IBM 2760 Optical Image Unit for the Insurance Industry (slides)	NEW
V20-0181-0	IBM CALL/360: BASIC (slides)	NEW
V20-0184-0	IBM 1288 Optical Page Reader (slides)	NEW
V20-4627-0	"For Your Information" (movie)	NEW
V20-6063-0	IBM CALL/360: BASIC (flipchart)	NEW
X20-1772-0	IBM 402, 403, 407 and 419 Accounting Machine Instructions	Use X24-6296 & X24-6300-1
Y20-0213-0	Student Scheduling System/360 The Scheduler Program (360A-US-07X) System Manual	NEW
Y20-0217-0	TNL Re: Y20-0161-0 Service for Consultants Update	NEW
Y20-0222-1	Insert (less Programming)	NEW
Y20-0223-1	Inserts (Programming only)	NEW
Y20-0224-0	IBM Managers Manual Binder, Index Tabs, and Corporate Inserts	NEW
Y20-0225-0	DP/FE/IR/OP Inserts Managers Manual	NEW
Y26-3726-0	IBM 1800 Multiprogramming Executive Operating System PLM	NEW
Y26-8015-0	IBM S/360 Disk and Tape Operating System Assembler (D) Re: Y26-3642-1 with Y26-8003 & Y26-8007	NEW

August 23, 1968 Issue No. 68-16

IBM INSTALLATION NEWSLETTER

Table of Contents

OS/360 DATA RETRIEVAL FROM DASD - CLARIFICATION

Installation Newsletter 68-06 contained an article, on page 4, on the above subject. The following should be inserted just before the discussion of the "Other procedures that may improve performance are: page 4.

> "Caution: When creating a Fixed or Fixed Block data set, it is possible for the last record, if truncated, to be written on the same track as normal length records. If the data set is then processed using Fixed-Block-Standard, the last record becomes an extra record on a track. The Access Method will not be looking for this record, and, therefore, the last record of the data set will be lost. To avoid this possibility, it is suggested that, instead of truncating the last record, it should be padded so that record uniformity may be maintained throughout the data set."

OS/360 COBOL F ISAM FILE ERROR HANDLING

The following is a field contribution which has not been submitted to any formal IBM test. Potential users should evaluate its usefulness in their own environment prior to implementation. No maintenance can be expected. As the author states, the subroutine has been used successfully under OS/360 Release 13. Release 14 has since become available and Release 15/16 announced with availability in August, 1968. Changes to COBOL F in future releases may affect the use of this approach.

When using OS/360 COBOL F to process ISAM files, two mutually exclusive options are available to the COBOL programmer for handling errors. This subject is discussed in the COBOL F Programmer's Guide (C28-6380). This reference does not adequately discuss the the subject. Two problems involved in using the declarative option follow. First, COBOL does not pass the DECB address for a BISAM file, but the DCB address. The DECB has a Hex Displacement of 100 past the DCB if the declarative section is used. Second, if the BISAM file is opened as an I-O file, a read statement which results in a record not found condition prevents a record from being added to the file until a successful read has been made followed by a rewrite. After a read, COBOL stores the record address from the logical record field of the DECB in a BLI cell. It then loads the file's base register from this cell. On a no-record-found ondition, this address consists of binary zeros. Therefore, no record can be constructed for this file.

Exhibit 1 contains an example of a subroutine approach which serves two functions. It is called by the declaratives section for both BISAM and QISAM file error processing. The standard errors are determined and set a condition code for COBOL to test. In addition, if a no-record-found condition is determined for a BISAM file, it obtains core for the record and modifies COBOL's BLI cell and base register before returning to COBOL.

This subroutine has been used successfully under OS/360 Release 13.

OS/360 COBOL F ISAM FILES ERROR HANDLING

```
*** THIS ROUTINE IS CALLED BY COBOL FROM THE DECLARATIVES SECTION TO ** ANLY000C
*** PROCESS ERRORS ON ISAM FILES. IT HAS TWO ENTRY POINTS, STORE AND ** ANLY0005
*** ANALYSIS. THE CALL FROM THE COBOL PROGRAM MUST BE CODED AS SHOWN ** ANLY0010
*** BELOW, DEPENDING ON THE MODE OF PROCESSING.                      ** ANLY0015
*                                                                     * ANLY002C
* FOR QISAM FILES ONLY                                               * ANLY0025
*        USE AFTER ERROR ON FILE-NAME                                * ANLY0030
*        ENTER LINKAGE                                               * ANLY0035
*        CALL @STORE@                                                * ANLY004C
*        ENTER COBOL                                                 * ANLY0045
*        ENTER LINKAGE                                               * ANLY0050
*        CALL @ANALYSIS@ USING FILE-NAME, PARAMETER1                 * ANLY0055
*        ENTER COBOL                                                 * ANLY006C
* WHERE FILE-NAME IS THE NAME ON THE COBOL FD STATEMENT AND           * ANLY0065
*       PARAMETER1 IS A 1 BYTE ALPHA-NUMERIC FIELD IN WORKING STORAGE * ANLY0070
*                                                                     * ANLY0075
* FOR BISAM FILES ONLY                                               * ANLY0080
*        USE AFTER ERROR ON FILE-NAME                                * ANLY0085
*        ENTER LINKAGE                                               * ANLY0090
*        CALL @ANALYSIS@ USING READ-STMNT PARAMETER1                 * ANLY0095
*        ENTER COBOL                                                 * ANLY010C
* WHERE READ-STMNT IS THE NAME OF A PARAGRAPH IN WHICH THE FIRST      * ANLY0105
*       STATEMENT IS READ FILE-NAME. FILE-NAME IS THE BISAM FILE.     * ANLY0110
*       PARAMETER1 IS THE SAME AS FOR QISAM FILES DISCUSSED ABOVE.    * ANLY0115
*                                                                     * ANLY012C
* CODES RETURNED TO COBOL IN PARAMETER1 ARE AS FOLLOWS                * ANLY0125
*  CODE      DESCRIPTION                    COBOL ACTION              * ANLY0130
*   1      LOWER LIMIT ON SETL NOT FOUND  CHECK KEY AND RETRY        * ANLY0135
*   2      SPACE NOT AVAILABLE            CLOSE FILES, TERMINATE     * ANLY014C
*   3      INVALID REQUEST                CLOSE FILES, TERMINATE     * ANLY0145
*   4      UNCORRECTIBLE INPUT ERROR      CLOSE FILES, TERMINATE     * ANLY0150
*   5      UNCORRECTIBLE OUTPUT ERROR     CLOSE FILES, TERMINATE     * ANLY0155
*   6      UNREACHABLE BLOCK              CLOSE FILES, TERMINATE     * ANLY016C
*   7      OVERFLOW RECORD                AS REQUIRED BY COBOL LOGIC * ANLY0165
*   8      SEQUENCE CHECK                 CHECK KEY, RETRY, ETC        ANLY0170
*   9      DUPLICATE RECORD               AS REQUIRED BY COBOL LOGIC * ANLY0175
*   A      NO RECORD FOUND                AS REQUIRED BY COBOL LOGIC * ANLY0180
*   B      RLI CHECK                    INVALID RLI, IF CORRECT ABEND* ANLY0185
*   C      DCB CLOSED                     NO FILE OPERATIONS ALLOWED * ANLY0190
* BLANK    NO ERROR INDICATORS SET        SHOULD NOT OCCUR SAFETY    * ANLY0195
*** FOR BISAM FILES RESULTING IN CODES 3,4,5,6, AND B, THE DECB FOR *** ANLY020C
*** THE FILE WILL BE DISPLAYED ON THE CONSOLE. FOR QISAM FILES      *** ANLY0205
*** RESULTING IN CODES 3,4,5,6, AND C, THE LOCATION OF THE STATUS   *** ANLY0210
*** BYTES, BUFFER AND DCB WILL BE DISPLAYED ON THE CONSOLE. THESE   *** ANLY0215
*** DISPLAYS USED IN CONJUNCTION WITH A DUMP WILL AID IN DEBUGGING  *** ANLY0220
         ISEQ     73,80                                              ANLY030C
STORE    START    0                                                  ANLY0305
         SAVE     (14,12)                STANDARD SAVE               ANLY0310
         USING    STORE,15               ESTABLISH ADDRESSIBILITY    ANLY0315
         STM      0,1,SAVE               STORE SYNAD RETURN REGISTERS ANLY0320
         MVI      SAVE,C'@'              '                           ANLY0325
         BR       14                     RETURN TO COBOL             ANLY033C
ANALYSIS SAVE     (14,12)                STANDARD SAVE               ANLY0335
         USING    ANALYSIS,15            ESTABLISH ADDRESSIBILTIY    ANLY0340
         L        REG3,STOREADR          POINT REG3 TO FIRST INSTRUCDION ANLY0345
         USING    STORE,3                ESTABLISH ADDRESSIBILITY    ANLY0350
         DROP     15                     DROP PREVIOUS BASE REGISTER ANLY0355
         ST       REG13,SAVEBLK+4        STORE HIGH SAVE IN SUBROUTINE ANLY0360
         MVC      8(4,13),ASAVEBLK       STORE SUBROUTINE SAVE IN HI SAVE ANLY0365
         LA       REG13,SAVEBLK          POINT TO SUBROUTINE SAVE    ANLY037C
```

Exhibit 1

OS/360 COBOL F ISAM FILES ERROR HANDLING

```
         B        BEGIN               START PROGRAM                   ANLY0375
SAVEBLK  DC       18F@0@              SAVE AREA                       ANLY0380
ASAVEBLK DC       A%SAVEBLK□                                          ANLY0385
SAVE     DC       2F@C@               SYNAD REG 0 AND 1 SAVE AREA     ANLY0390
STOREADR DC       A%STORE□                                            ANLY0395
REG0     EQU      0                                                   ANLY0400
REG1     EQU      1                                                   ANLY0405
REG2     EQU      2                                                   ANLY0410
REG3     EQU      3      PROGRAM@S BASE REGISTER                      ANLY0415
REG4     EQU      4                                                   ANLY0420
REG5     EQU      5      POINTS TO DCB OR READ STATEMENT              ANLY0425
REG6     EQU      6      POINTS TO RETURN CODE                        ANLY0430
REG7     EQU      7                                                   ANLY0435
REG8     EQU      8                                                   ANLY0440
REG9     EQU      9                                                   ANLY0445
REG10    EQU      10                                                  ANLY0450
REG11    EQU      11                                                  ANLY0455
REG12    EQU      12                                                  ANLY0460
REG13    EQU      13                                                  ANLY0465
REG14    EQU      14                                                  ANLY0470
REG15    EQU      15                                                  ANLY0475
         ENTRY    ANALYSIS                                            ANLY0480
BEGIN    LM       REG5,REG6,0%REG1□   LOAD PASSED PARAMETER ADDRESSES ANLY0500
         CLI      SAVE,C@Q@           WAS STORE ENTERED               ANLY0505
         BE       QISAM               YES, FILE IS QISAM              ANLY0510
BISAM    L        REG1,SAVEBLK+4      GET REG13 ENTRY VALUE           ANLY0515
         MVC      EXEC+2%2□,2%5□                                      ANLY0520
         NI       EXEC+2,X@0F@                                        ANLY0525
         OI       EXEC+2,X@10@                                        ANLY0530
         LR       REG4,REG5           HOLD POINTER TO READ            ANLY0531
EXEC     L        REG5,0%0,REG1□                                      ANLY0535
* THE ABOVE 5 INSTRUCTIONS RESULT IN REGISTER 5 CONTAINING THE ADDRESS ANLY0540
* OF THE DECB. REGISTER 5 ORIGINALLY POINTED TO A COBOL READ STATEMENT ANLY0545
* WHICH GENERATED A  L  REGX,DISP%0,REG13□ INSTRUCTION. THE ABOVE MVC  ANLY0550
* INSTRUCTION MOVES THE SECOND OPERAND TO THE SECOND OPERAND OF THE    ANLY0555
* EXEC INSTRUCTION. THE AND AND OR INSTRUCTINS CHANGE THE BASE REG TO  ANLY0560
* REGISTER 1 WHICH WAS LOADED WITH VALUE COBOL HAD IN REGISTER 13.     ANLY0565
         TM       24%REG5□,X@80@      TEST FOR RECORD NOT FOUND       ANLY0570
         MVI      0%REG6□,C@A@        SET RECORD NOT FOUND CODE       ANLY0575
         BO       CHANGE              IF SO, GO TO GETMAIN ROUTINE    ANLY0580
         TM       24%REG5□,X@08@      TEST FOR UNCORRECTABE I-O ERROR ANLY0585
         BO       TESTTYPE            IS SO, TEST FOR I OR O          ANLY0590
         TM       24%REG5□,X@04@      TEST FOR UNREACHABLE BLOCK      ANLY0595
         MVI      0%REG6□,C@6@        SET UNREACHABLE BLOCK CODE      ANLY0600
         BO       PRINT               IF SO, GO TO PRINT DECB ROUTINE ANLY0605
         TM       24%REG5□,X@01@      TEST FOR DUPLICATE RECORD       ANLY0610
         MVI      0%REG6□,C@9@        SET DUPLICATE RECORD CODE       ANLY0615
         BO       RETURN              IF SO, RETURN TO COBOL          ANLY0620
         TM       24%REG5□,X@20@      TEST FOR SPACE NOT FOUND        ANLY0625
         MVI      0%REG6□,C@2@        SET SPACE NOT FOUND CODE        ANLY0630
         BO       RETURN              IF SO, RETURN TO COBOL          ANLY0635
         TM       24%REG5□,X@40@      TEST FOR RLI CHECK              ANLY0640
         MVI      0%REG6□,C@B@        SET RLI CHECK CODE              ANLY0645
         BO       PRINT               IF SO, GO TO PRINT DECB ROUTINE ANLY0650
         TM       24%REG5□,X@10@      TEST FOR INVALID REQUEST        ANLY0655
         MVI      0%REG6□,C@3@        SET INVALID REQUEST CODE        ANLY0660
         BO       PRINT               IF SO, GO TO PRINT DECB ROUTINE ANLY0665
         TM       24%REG5□,X@02@      TEST FOR OVERFLOW RECORD        ANLY0670
         MVI      0%REG6□,C@7@        SET OVERFLOW RECORD CODE        ANLY0675
         BO       RETURN              IF SO, RETURN TO COBOL          ANLY0680
```

Exhibit 1 (Continued)

OS/360 COBOL F ISAM FILE ERROR HANDLING

```
         MVI    0%REG6¤,C@ @          SET NO ERRORS FOUND CODE          ANLY0685
         B      RETURN                RETURN TO COBOL                   ANLY0690
TESTTYPE TM     5%REG5¤,X@A0@         TEST DECBTYP2 FOR READ OPERATION  ANLY0700
         MVI    0%REG6¤,C@4@          SET UNCORRECTABLE INPUT ERROR     ANLY0705
         BM     PRINT                 IF SO, GO TO PRINT DECB ROUTINE   ANLY0710
         MVI    0%REG6¤,C@5@          SET UNCORRECTABLE OUTPUT ERROR    ANLY0715
PRINT    BCTR   REG5,0                DECREMENT DECB POINTER            ANLY0720
         LA     REG7,26               ESTABLISH DECB BYTE COUNT         ANLY0725
         LA     REG8,51               ESTABLISH OUTPUT INDEX VALUE      ANLY0730
         LA     REG9,WTO              POINT TO OUTPUT AREA              ANLY0735
         LA     REG12,XLATE                                             ANLY0736
         SR     REG10,REG10           CLEAR REGISTER                    ANLY0740
INSERT   IC     REG10,0%REG7,REG5¤    PICK UP RIGHT BYTE OF DECB        ANLY0745
         LA     REG6,2                ESTABLISH SHIFT COUNT             ANLY0750
SHIFT    SRDL   REG10,4               SHIFT LOW ORDER HALF BYTE TO 11   ANLY0755
         SRL    REG11,28              RIGHT ADJUST REG11                ANLY0760
         STC    REG11,0%REG8,REG9¤    PLACE IN OUTPUT AREA              ANLY0765
         BCTR   REG8,0                DECREMENT INDEX                   ANLY0770
         BCT    REG6,SHIFT            DO HIGH ORDER HALF BYTE           ANLY0775
         BCT    REG7,INSERT           ANY MORE TO SHIFT                 ANLY0780
         BR     REG12                                                   ANLY0781
XLATE    TR     WTO,TABLE             NO, TRANSLATE OUTPUT              ANLY0785
         WTO    MF#%E,MSG1¤           PUT OUT MESSAGE                   ANLY0790
         MVI    WTO,X@00@             CLEAR MESSAGE                     ANLY0795
         MVC    WTO+1%51¤,WTO         AREA                              ANLY0800
         B      RETURN                RETURN TO COBOL                   ANLY0805
CHANGE   L      REG2,8%0,REG4¤        GET DCB ADDRESS                   ANLY0810
         L      REG2,80%0,REG2¤       GET LRECL FIELD FROM DCB          ANLY0815
         SLL    REG2,16               CLEAR OUT 2 GARBAGE BYTES         ANLY0820
         SRL    REG2,19               AND DIVIDE BY 8                   ANLY0825
         A      REG2,#F@3@            ADD 3 TO INSURE ADEQUATE SPACE    ANLY0830
         SLL    REG2,3                MULTIPLY BY 8                     ANLY0835
         C      REG2,SIZE             WAS LAST GETMAIN LARGE ENOUGH     ANLY0840
         BNH    ALTER                 YES                               ANLY0845
         CLC    CORELOC,#F@0@         HAS A GET MAIN BEEN ISSUED        ANLY0850
         BE     GETMAIN               NO                                ANLY0855
         L      REG1,CORELOC          LOAD PREVIOUS GETMAIN ADDRESS     ANLY0860
         L      REG0,SIZEASK          LOAD PREVIOUS GETMAIN SIZE        ANLY0865
         FREEMAIN R,LV#%0¤,A#%1¤      FREEMAIN                          ANLY0870
GETMAIN  L      REG0,REG2             PLACE REQUESTED SIZE IN REG0      ANLY0875
         GETMAIN R,LV#%0¤             ISSUE GETMAIN                     ANLY0880
         ST     REG1,CORELOC          STORE ADDRESS OF AREA             ANLY0885
         ST     REG0,SIZEASK          STORE SIZE OF AREA                ANLY0890
         A      REG1,#F@16@           POINT TO CORE RETURNED PLUS 16    ANLY0900
         ST     REG1,LOC              SAVE LOCATION FOR COBOL           ANLY0905
         S      REG0,#F@16@           SUBTRACT 16 BYTES FROM SIZE       ANLY0910
         ST     REG0,SIZE             STORE REDUCED SIZE                ANLY0915
ALTER    MVC    COBCLST+2%2¤,36%REG4¤                                   ANLY0920
         NI     COBOLST+2,X@0F@                                         ANLY0925
         OI     COBOLST+2,X@10@                                         ANLY0930
         L      REG1,SAVEBLK+4                                          ANLY0935
         L      REG0,LOC                                                ANLY0940
COBCLST  ST     REG0,0%0,REG1¤                                          ANLY0945
* THE ABOVE 6 INSTRUCTIONS RESULT IN THE GETMAIN ADDRESS BEING STORED    ANLY0950
* IN THE COBOL PROGRAM@S BLI CELL. REG4 POINTS TO THE COBOL READ         ANLY0955
* STATEMENT. THE 34TH BYTE OF THIS EXPANSION IS THE START OF A           ANLY0960
* ST REGX,DISP%C,REG13¤ INSTRUCTION. THE ABOVE MVC INSTRUCITON MOVES     ANLY0965
* THE BASE AND DISPLACEMENT TO THE COBOLST INSTUUCTION. THE NI AND OI    ANLY0970
* INSTRUCITONS CHANGE THE BASE REGISTER TO REG1 WHICH WAS LOADED WITH    ANLY0975
* THE VALUE COBCL HAD IN REGISTER 13.                                    ANLY0980
```

Exhibit 1 (Continued)

OS/360 COBOL F ISAM FILE ERROR HANDLING

```
         SR        REG2,REG2              CLEAR REGISTER                    ANLY1000
         IC        REG2,39%0,REG4¤        GET BASE REG OF COBOL@S LOAD      ANLY1005
         SRL       REG2,4                 SHIFT OFF INDEX REG               ANLY1010
         SLL       REG2,2                 MULTIPLY BY 4                     ANLY1015
         C         REG2,#F@48@            WAS BASE REG OVER 12              ANLY1016
         BNH       STOREIT                NO                                ANLY1017
         B         RETURN                 IF OVER 12, THE VALUE CANNOT BE   ANLY1018
* STORED. IT IS ASSUMED COBOL WILL LOAD REG14 & 15 FROM THE BLI CELL         ANLY1019
* WHEN THE COMPILER IS FORCED TO ASSIGN REG14 & REG15 ON A TEMPORARY        ANLY1020
* BASIS                                                                     ANLY1021
STOREIT  ST        REG0,20%REG2,REG1¤     STORE LOC ADDR IN CORRECT SAVE    ANLY1022
* THE ABOVE 5 INSTRUCTIONS CHANGE THE COBOL FILE@S BASE REGISTER IN         ANLY1025
* THE COBOL SAVE AREA SO THAT IT IS LOADED PROPERLY ON RETURN. THE IC       ANLY1030
* INSTRUCTION PICKS UP THE BASE REG FROM THE COBOL INSTRUCTION              ANLY1035
* L BASEREG,DISP%0,REG13¤. THE INDEX REGISTER IS SHIFTED OFF. THE BASE      ANLY1040
* REG DESIGNATION IS MULTIPLIED BY 4 AND REG2 BECOMES AN INDEX REG.         ANLY1045
* REG1 POINTS TO THE COBOL SAVE AREA. THE 20 DISPLACEMENT POINTS TO         ANLY1050
* REG0 SAVE LOCATION                                                        ANLY1055
         B         RETURN                                                   ANLY1060
QISAM    MVI       SAVE,X@00@             CLEAR QISAM INDICATOR             ANLY1100
         TM        81%REG5¤,X@80@         TEST FOR SEQUENCE CHECK           ANLY1105
         MVI       0%REG6¤,C@8@           SET SEQUENCE CHECK CODE           ANLY1110
         BO        RETURN                 RETURN TO COBOL                   ANLY1115
         TM        80%REG5¤,X@20@         TEST FOR SPACE NOT FOUND          ANLY1120
         MVI       0%REG6,C@2@            SET SPACE NOT FOUND CODE          ANLY1125
         BO        RETURN                 RETURN TO COBOL                   ANLY1130
         TM        81%REG5¤,X@40@         TEST FOR DUPLICATE RECORD         ANLY1135
         MVI       0%REG6¤,C@9@           SET DUPLICATE RECORD CODE         ANLY1140
         BO        RETURN                 RETURN TO COBOL                   ANLY1145
         TM        80%REG5¤,X@08@         TEST FOR INPUT ERROR              ANLY1150
         MVI       0%REG6¤,C@4@           SET INPUT ERROR CODE              ANLY1155
         BO        PRINT2                 PRINT ADDRESSES                   ANLY1160
         TM        80%REG5¤,X@04@         TEST FOR OUTPUT ERROR             ANLY1165
         MVI       0%REG6¤,C@5@           SET OUTPUT ERROR CODE             ANLY1170
         BO        PRINT2                 PRINT  ADDRESSES                  ANLY1175
         TM        80%REG5¤,X@C0@         TEST FOR LOWER LIMIT NOT FOUND    ANLY1180
         MVI       0%REG6¤,C@1@           SET LOWER LIMIT NOT FOUND CODE    ANLY1185
         BM        RETURN                                                   ANLY1190
         TM        80%REG5¤,X@02@         TEST FOR UNREACHABLE BLOCK        ANLY1200
         MVI       0%REG6¤,C@6@           SET UNREACHABLE BLOCK CODE        ANLY1205
         BO        PRINT2                 PRINT ADDRESS ROUTINE             ANLY1210
         TM        80%REG5¤,X@10@         TEST FOR INVALID REQUEST          ANLY1215
         MVI       0%REG6¤,C@3@           SET INVALID REQUEST CODE          ANLY1220
         BO        PRINT2                 PRINT ADDRESS ROUTINE             ANLY1225
         TM        81%REG5¤,X@20@         TEST FOR DCB CLOSED               ANLY1230
         MVI       0%REG6¤,C@C@           SET DCB CLOSED CODE               ANLY1235
         BO        PRINT2                 PRINT ADDRESS ROUTINE             ANLY1240
         TM        81%REG5¤,X@10@         TEST FOR OVERFLOW RECORD          ANLY1245
         MVI       0%REG6¤,C@7@           SET OVERFLOW RECORD CODE          ANLY1250
         BO        RETURN                 RETURN TO COBOL                   ANLY1255
         MVI       0%REG6¤,C@ @           SET NO ERROR FOUND CODE           ANLY1260
         B         RETURN                                                   ANLY1265
PRINT2   LA        REG9,STATUS            POINT TO OUTPUT                   ANLY1300
         LA        REG7,3                 ESTABLISH BYTE COUNT              ANLY1305
         ST        REG5,SAVEBLK           SAVE DCB ADDRESS                  ANLY1306
         LA        REG5,SAVE              POINT TO REG0 SAVE FIELD          ANLY1310
         LA        REG8,5                 ESTABLISH OUTPUT INDEX COUNT      ANLY1315
         BAL       REG12,INSERT           GO TO CONVERSION ROUTINE          ANLY1320
         LA        REG9,BUFFER            POINT TO OUTPUT                   ANLY1325
         LA        REG7,3                 ESTABLISH BYTE COUNT              ANLY1330
```

Exhibit 1 (Continued)

OS/360 COBOL F ISAM FILE ERROR HANDLING

```
            LA      REG5,SAVE+4          POINT TO REG1 SAVE FIELD          ANLY1335
            LA      REG8,5               ESTABLISH OUTPUT INDEX COUNT      ANLY1340
            BAL     REG12,INSERT         GO TO CONVERSION ROUTINE          ANLY1345
            LA      REG9,DCB             POINT TO OUTPUT                   ANLY1350
            LA      REG7,3               ESTABLISH BYTE COUNT              ANLY1355
            LA      REG5,SAVEBLK         POINT TO STORED DCB ADDRESS       ANLY1360
            LA      REG8,5               ESTABLISH OUTPUT INDEX COUNT      ANLY1365
            BAL     REG12,INSERT         GO TO CONVERSION ROUTINE          ANLY1370
            TR      STATUS,TABLE         CHANGE THE THREE                  ANLY1375
            TR      BUFFER,TABLE         OUTPUT FIELDS TO                  ANLY1380
            TR      DCB,TABLE            A HEX FORMAT                      ANLY1385
            WTO     MF#%E,MSG2□                                            ANLY1390
            B       RETURN               GO TO COBOL                       ANLY1395
TABLE       DC      C@0123456789ABCDEF@                                    ANLY1400
            DS      0F                                                     ANLY1405
MSG1        DC      AL2%MSG1END-MSG1□                                      ANLY1410
            DC      H@0@                                                   ANLY1415
            DC      C@DECB#@                                               ANLY1420
WTO         DC      52X@00@                                                ANLY1425
MSGEND1     EQU     *                                                      ANLY1430
SIZE        DC      F@0@    WILL CONTAIN THE GETMAIN BYTE SIZE MINUS 16    ANLY1445
CORELOC     DC      F@0@    WILL CONTAIN THE GETMAIN CORE LOCATION         ANLY1450
SIZEASK     DC      FC@     WILL CONTAIN THE GETMAIN BYTE SIZE             ANLY1455
LCC         DC      F@0@    WILL CONTAINS THE GETMAIN CORE LOCATION PLUS 16ANLY1460
MSG2        DC      AL2%MSG2END-MSG2□                                      ANLY1465
            DC      H@0@                                                   ANLY1470
            DC      C@STATUS BYTES AT @                                    ANLY1475
STATUS      DC      CL6@ @                                                 ANLY1480
            DC      C@ BUFFER AT @                                         ANLY1485
BUFFER      DC      CL6@ @                                                 ANLY149C
            DC      C@ DCB AT @                                            ANLY1495
DCB         DC      CL6@ @                                                 ANLY1496
MSG2END     EQU     *                                                      ANLY1497
            END     STORE                                                  ANLY1499
```

Exhibit 1 (Continued)

OS/360 MFT-II AND MVT LABELED TAPE CONSIDERATIONS

Labeled Tapes

The following should be considered when planning the installation of a multijobbing system in a fully labeled installation.

1. Labeled data tapes used by jobs are part of the criteria for establishing job classes in this environment. Indiscriminate assignments can result in performance degradation due to concurrent jobs requesting the same tape volume. As a result, one or more of these jobs will have to wait for that tape to be released.

2. The same situation can exist with labeled scratch tapes if programmers are allowed to call for them by specific volume serial number, i.e., two concurrent jobs in two partitions both requesting a labeled scratch tape by VOLUME=SER=001.

DOS/360 CATALOGING COMPILER OUTPUT TO RELOCATABLE LIBRARY

The following is from the IBM WTC Field Support Center, Sydney, Australia. It has not been submitted to any formal IBM test. Potential users should evaluate its usefulness in their own environment prior to implementation.

A convenient program testing technique consists of assigning SYSPCH to disk or tape and putting object programs onto magnetic storage. SYSIPT can then be assigned to the tape or disk extent concerned, and the object programs may be cataloged to the relocatable library without being punched into cards.

However, for this technique to work, a CATALR modname card is required immediately preceding the object deck. Techniques for producing this card DOS/360 are outlined below.

ASSEMBLER - use a PUNCH statement in the assembly.

PL/I - Set UPSI Switch 0 on (Version 3).

COBOL, RPG, FORTRAN - No direct facility exists.

The most common solution in this case is to execute a small assembly immediately preceding the compilation: e.g.

```
// EXEC ASSEMBLY
  PUNCH ' CATALR modname'
  END
/*
```

DOS/360 - EXECUTION OF 14K OR 44K ASSEMBLER IN BG

The following technique is a field contribution which has not been submitted to any formal IBM test. Potential users should evaluate its usefulness in their own environment prior to implementation. No maintainance can be expected.

Some DOS/360 users with a 64K S/360 Model 30 when operating in a multiprogramming environment may not be able to use the 44K Assembler (Release 17) because of the size of the background partition. The following technique allows execution of the 14K or 44K Assembler depending upon the size of the background partition.

To obtain maximum throughput on Assemblies, both the 14k variant of the Assembler and the 44K Assembler should be cataloged to the Core Image Library. The former can be used when multiprogramming (i.e. when the background area is less than 44K) and the later can be used when not multiprogramming (i.e. when the background area is greater than 44K).

In order to continue using the // EXEC ASSEMBLY job control statement;

1. The module IJQD32 should be punched from the Relocatable Library (private or system) and re-cataloged with the phase name in the second card changed from ASSEMBLY to ASSEMB14.

2. Likewise, the module IJYASM should be punched from the private Relocatable Library and re-cataloged with the phase name changed to ASSEMB44.

3. IJQD32 and IJYASM should be Link-Edited and cataloged to the Core Image Library.

4. The subroutine shown in Exhibit 1 should be cataloged to the Core Image Library with the phase name ASSEMBLY. It will interrogate the Communication Region for the size of the background area and FETCH the correct Assembler.

DOS/360 COBOL USE OF OCCURS AND REDEFINES

The restrictions on the use of OCCURS with REDEFINES, as stated in the SRL "DOS/TOS COBOL Language Specifications" (C24-3433-4), page 58, have been and will continue to be in effect. These also apply to OS/360 COBOL E. In earlier releases, most of the illegal combinations of OCCURS and REDEFINES were not diagnosed, and, if diagnosed, a warning message was issued. In DOS/360 Release 17, a correction was made to issue an E-level diagnostic if a REDEFINES clause appears in an item immediately subordinate to an item containing an OCCURS clause. For planning purposes you are advised that present plans are that a subsequent release will diagnose the use of a REDEFINES clause in all levels subordinate to an item with an OCCURS clause. This change, if made, will be when and as provided through the normal IBM documentation procedures.

For users who will be impacted by this correction, the following suggestion may be helpful. Note that this is a recommended programming technique, in any case, for improving program efficiency when dealing with subscripted data items.

Define, in Working Storage, an image record of the OCCURS portion of the master-record. The image record may contain any necessary redefinitions. Then move the OCCURS item to the image record and perform any necessary operations, working with non-subscripted data items.

For example, see Exhibit 1.

DOS/360-EXECUTION OF 14K OR 44K ASSEMBLER IN BG

DOS CL3-1 08/01/68

```
LOC    OBJECT CODE     ADDR1 ADDR2  STMT   SOURCE STATEMENT
004000                               1        START 16384
004000 0520                          2 BEGIN  BALR  2,0
004002                               3        USING *,2
                                     4 *
                                     5 * THE COMMUNICATION REGION CONTAINS THE UPPERMOST BYTE OF THE BG AREA
                                     6 * IN LOCATION 32 AND THE FIRST BYTE OF THE BG AREA IN LOCATION 10
                                     7 *
                                     8        COMRG
                                     9+** 360N-CL-453 COMRG     CHANGE LEVEL 3-0
004002 5810 0014             00014  10+      L     1,20
004006 0A21                         11+      SVC   33
004008 5830 1020             00020  12       L     3,32%0,1¤ LOAD ADDRESS OF UPPERMOST BYTE OF BG AREA
00400C 4B30 100A             0000A  13       SH    3,10%0,1¤ SUBTRACT ADDRESS OF FIRST BYTE OF BG AREA
004010 5930 2036             04038  14       C     3,#F@45055@ COMPARE RESULT TO SIZE OF 44K ASSEMBLER
004014 4740 201E             04020  15       BL    ASSEMB14                                  MINUS ONE
                                    16       FETCH ASSEMB44
                                    17+** 360N-CL-453 FETCH     CHANGE LEVEL 3-0
004018                              18+      DC    0H@0@
004018 4110 2026             04028  19+      LA    1,#CL8@ASSEMB44@
00401C 1B00                         20+      SR    0,0
00401E 0A01                         21+      SVC   1
                                    22 ASSEMB14 FETCH ASSEMB14
                                    23+** 360N-CL-453 FETCH     CHANGE LEVEL 3-0
004020                              24+ASSEMB14 DC    0H@0@
004020 4110 202E             04030  25+      LA    1,#CL8@ASSEMB14@
004024 1B00                         26+      SR    0,0
004026 0A01                         27+      SVC   1
004000                              28       END   BEGIN
004028 C1E2E2C5D4C2F4F4             29             #CL8@ASSEMB44@
004030 C1E2E2C5D4C2F1F4             30             #CL8@ASSEMB14@
004038 0000AFFF                     31             #F@45055@
```

Exhibit 1

DOS/360 COBOL USE OF OCCURS AND REDEFINES

```
(FD in File Section)

        Ø1 REC.
              Ø2 PART-1 OCCURS 18 PICTURE X (20).
                    .
                    .
                    .


WORKING-STORAGE SECTION.

        Ø1 REC-IMAGE.
              Ø2 A PICTURE S9 (5) COMPUTATIONAL-3.
              Ø2 B REDEFINES A PICTURE X (3).
              Ø2 C.
                    Ø3 D PICTURES 9.
                    Ø3 E REDEFINES D PICTURE X.
                          .
                          .
                          .


PROCEDURE DIVISION.
                    .
                    .
                    .
        PERFORM PARA VARYING N FROM 1 BY 1
        UNTIL N GREATER THAN 18.
                    .
                    .
PARA.  MOVE PART-1 (N) TO REC-IMAGE.
                    .
                    .
                    .
        (coding to perform operations using A, B, D, E etc.)
                    .
                    .
```

Exhibit 1

DOS/360 DISK EXTENTS FOR COBOL PROGRAMS SRL CORRECTION

The DOS/360 COBOL Programmer's Guide C24-5025-4 is in error on page 26 in discussing the use of the new DLBL and EXTENT cards for DOS.

The 'filename' in the //DLBL statement must be in SYS nnn format as was necessary in previous VOL statements, not the 1 to 7 characters as the manual indicates.

The SRL will be corrected with the next TNL or revision.

DOS/360 RELEASE 17 RPG - NEW ERROR CODES

DOS/360 Release 17 contained ISAM load and add facilities in RPG. The TNL Y21-0005 to the PLM Y26-3701-1 contains seven new error codes which may have caused a program to terminate as a result of H0. Users who do not receive the PLM's should know of these additions which are contained in the updated H0 chart shown in Exhibit 1.

HALT INDICATOR (H0) ANALYSIS AID

Displacements in Decimal From GR3	284	288	289	290
Condition That Turned H0 On	*	Resulting Byte Combinations Set (Hexadecimal)		
Initialized on or on due to programmer request	N/A	00	00	00
Invalid chaining request	(A)	02	N/A	N/A
Undefined record type	(B)	10	N/A	N/A
Collating sequence error (matching records)	N/A	04	N/A	N/A
Record sequence error (predetermined sequence)	N/A	08	N/A	N/A
DAM (record not found)	(C)	N/A	80	N/A
DAM (data check)	(C)	N/A	40	N/A
DAM (wrong length record)	(C)	N/A	20	N/A
ISAM (invalid key length)	(B)	N/A	N/A	FF
ISAM (DASD error)	(C)	N/A	N/A	80
ISAM (wrong length record)	(C)	N/A	N/A	40
ISAM (illegal EOF-within limits)	(C)	N/A	N/A	20
ISAM - Load (prime data area full)	(C)	N/A	N/A	20
ISAM - Load (master index full)	(C)	N/A	N/A	08
ISAM - (duplicate record)	(C)	N/A	N/A	04
ISAM - Load (sequence error)	(C)	N/A	N/A	02
ISAM - Add (overflow area full)	(C)	N/A	N/A	02
ISAM - (no record found)	(C)	N/A	N/A	10
ISAM - Load (cylinder index area full)	(C)	N/A	N/A	10

*

(A) = Chaining identifier
(B) = Address of IORB
(C) = Address of DTF table

Exhibit 1

DOS/360 RELEASE 18 PL/I PLANNING INFORMATION

The following information is provided for planning purposes. The actual availability and specifications will be as provided through the official IBM procedures, the Program Announcement and formal documentation.

1. With Release 18, the priority of the concatenation operator has been changed from level 7 to level 4. (Between INFIX + and the Logical Operators).

2. It will be necessary to read the various PL/I TNLs prior to designing a file with the new I/O facilities available in release 18 (as it was in the past). For example, if the rules for fixed length blocked records with READ SET or LOCATE statements (record size must be divisible by 8) were not noticed, a redesign of a system might have been necessary.

3. Version 3 of PL/I in DOS Release 18, also incorporates some important optimization improvements in the area of fixed decimal arithmetic and structure assignments. To take advantage of the fixed decimal improvements, declare all FIXED DECIMAL fields with an odd precision. Within each expression and/or assignment, keep the scale and precision of the operands the same. The receiving field should be large enough to accommodate the expression results (as determined by the implementation precision rules).

4. To take advantage of structure assignment optimization, the relative structuring and internal representation of the elements must be identical. (This optimization will not be effected with structure expressions.) A similar optimization for array assignments can be effected by declaring the array as a single element structure. For example, the following will generate and execute one MVC command where before it would have generated a loop executing 100 MVCs.

```
Declare
    1 A , 2 B (100) character (2),
    1 C , 2 D (100) character (2);
B = D ; /* generates 1 MVC. */
```

BPS/360 REDUCING CORE IMAGE LIBRARY SIZE FOR BAL

The following was developed by an IBM customer and contributed to the Newsletter. Potential users should be aware that it has not been submitted to any IBM test and should evaluate its usefulness in their own environment prior to implementation.

Installation Newsletter 68-03 contained, on page 3, an article concerned with reducing Core Image Library size for DOS/360 by storing repetitive type modules only one time in the CIL. After an IBM Systems Engineer showed it to one of his customers, they developed another idea for reducing the size of the CIL. The idea was not to store the I/O areas for BAL programs on the disk. Exhibit 1 illustrates the technique. Essentially what it amounts to is assigning core storage for I/O areas at execution time. This is particularly advantageous when large blocked records are used by the processing program.

BPS/360 REDUCING CORE IMAGE LIBRARY SIZE FOR BAL

How to save core image library when writting in B. A. L. , and to have the post list and any core dump to coincide.

Put the I/O areas immediately after the start statement as equate entries.

EXAMPLE: AREA1 EQU *-3602

Equate each I/O area to *-(the length of the I/O area PLUS the length of all other I/O areas already equated).

The last equated amount is divided by eight (8) and the difference between the remainder and eight (8) is added to the last equated amount to equal AMOUNTX. (This is to align on double word boundary).

AMOUNTX plus (length of supervisor - (8192)) plus (length of label checking area) is the amount to put in the start statement.

AMOUNTX is the amount to put in the PHASE card.

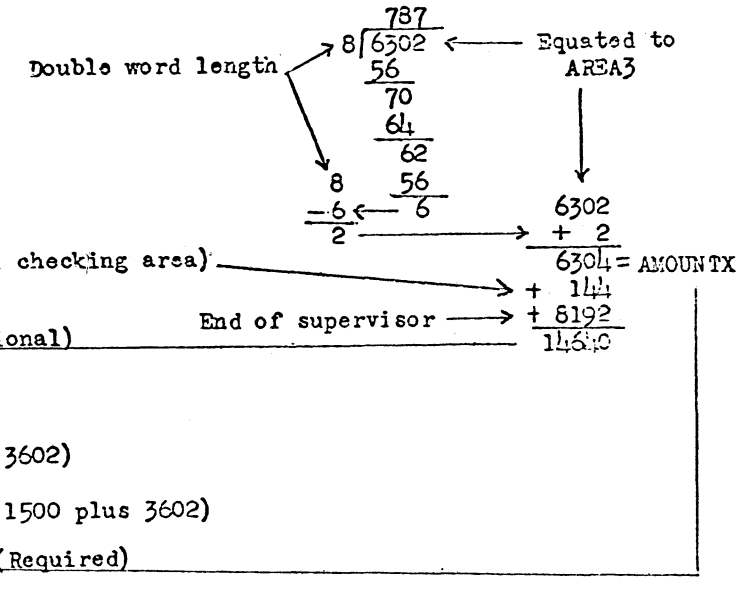
EXAMPLE:

AREA1 3602 bytes

AREA2 1500 bytes

AREA3 1200 bytes

// LBLTYP NSD(3) 144 bytes (label checking area)




```
        START   14640   <-- (Optional)
AREA1   EQU     *-3602
AREA2   EQU     *-5102 (1500 plus 3602)
AREA3   EQU     *-6302 (1200 plus 1500 plus 3602)
   PHASE progname,S+6304  <-- (Required)
```

Exhibit 1

S/360 20 DPS ADDRESSING OVER 4K, 8K, 12K OR 16K

The following is a field contribution which has not been submitted to any formal IBM test. Potential users should evaluate its usefulness in their own environment. No maintenance can be expected.

The purpose of the BSREG macro is to provide Model 20 Assembler language programmers the ability to address over 4K, 8K, 12K or 16K by means of one programmer-written statement. The format of that statement is described in Exhibit 1.

The BSREG macro definition is shown in Exhibit 2. Examples of its use including generations are shown in Exhibit 3.

S/360 20 DPS ADDRESSING OVER 4K, 8K, 12K OR 16K

NAME FIELD	OPERATION CODE	OPERANDS (IN ANY SEQUENCE)		
Blank or any valid Assembler Symbol	BSREG	Keyword	Allowable Specification	Remarks
		K=	4, 8, 12 or 16	Indicates core size in terms of multiples of 1,024 bytes. If omitted, 4 is assumed.
		R1=	Any valid Model 20 register (8-15).	Specifies the register to be used to address the first 4K. If omitted, 10 is assumed.
		R2=	Any valid Model 20 register (8-15).	Specifies the register to be used to address the second 4K. If omitted and K=8 or more, 11 is assumed.
		R3=	Any valid Model 20 register (8-15).	Specifies the register to be used to address the third 4K. If omitted and K=12 or more, 12 is assumed.
		R4=	Any valid Model 20 register (8-15).	Specifies the register to be used to address the last 4K. If omitted and K=16, 13 is assumed.

Exhibit I

S/360 DPS ADDRESSING OVER 4K, 8K, 12K OR 16K

The BSREG Macro Definition, Including the
DPS Job Control Cards Necessary to Catalog It.

**Programmer's Note: Be Certain the // END Card is processed! **
Otherwise you may clobber SYSRES in subsequent runs.

```
// LOG
// JOB MMAINT
// DATE 68277
// EXEC
MODEL 20 DPS -MACRO MAINTENANCE PROGRAM- VERSION 01  MOD-LEVEL 02

// CATAL

          MACRO

STMNT NO.  MACRO DEFINITION STATEMENT

           &NAME     BSREG &K#4,&R1#1C,&R2#11,&R3#12,&R4#13

BSREG001   &AL1      SETA  &K
BSREG002   &BL1      SETB  %&AL1 EQ 4¤
BSREG003             AIF   %&BL1¤.OK
BSREG004   &BL1      SETB  %&AL1 EQ 8¤
BSREG005             AIF   %&BL1¤.OK
BSREG006   &BL1      SETB  %&AL1 EQ 12¤
BSREG007             AIF   %&BL1¤.OK
BSREG008   &BL1      SETB  %&AL1 EQ 16¤
BSREG009             AIF   %NOT &BL1¤.ERR
BSREG010   .OK       ANOP
BSREG011   &NAME     BASR  &R1,0
BSREG012             USING *,&R1
BSREG013             AIF   %&AL1 EQ 4¤.LAST
BSREG014   A&SYSNDX  EQU   *
BSREG015             USING A&SYSNDX+4096,&R2
BSREG016             LH    &R2,B&SYSNDX
BSREG017             AIF   %&AL1 EQ 8¤.BC
BSREG018             USING A&SYSNDX+8192,&R3
BSREG019             LH    &R3,C&SYSNDX
BSREG020             AIF   %&AL1 EQ 12¤.BC
BSREG021             USING A&SYSNDX+12288,&R4
BSREG022             LH    &R4,D&SYSNDX
BSREG023             AIF   %&AL1 EQ 16¤.BC
BSREG024   .ERR      MNOTE 0,'INCORRECT SPECIFICATION OF CORE SIZE AS &AL1'
BSREG025             MEXIT

BSREG026   .BC       BC    15,E&SYSNDX
BSREG027   B&SYSNDX  DC    Y%A&SYSNDX+4096¤
BSREG028             AIF   %&AL1 GT 8¤.GEN12
BSREG029             AGO   .END
BSREG030   .GEN12    ANOP
BSREG031   C&SYSNDX  DC    Y%A&SYSNDX+8192¤
BSREG032             AIF   %&AL1 GT 12¤.GEN16
BSREG033             AGO   .END
BSREG034   .GEN16    ANOP
BSREG035   D&SYSNDX  DC    Y%A&SYSNDX+12288¤
BSREG036   .END      ANOP
BSREG037   E&SYSNDX  EQU   *
BSREG038   .LAST     MEND

// END
```

Exhibit 2

S/360 20 DPS ADDRESSING OVER 4K, 8K, 12K OR 16K

Examples of the use of the BSREG macro, including actual generations.

```
 TEST5    BSREG
*TEST5    BASR  10,0
*         USING *,10
```

(BSREG with label but no operands. Note: K=4 and R1=10 were assumed.)

```
 HERE     BSREG K#16
*HERE     BASR  10,0
*         USING *,10
*A0004    EQU   *
*         USING A0004+4096,11
*         LH    11,B0004
*         USING A0004+8192,12
*         LH    12,C0004
*         USING A0004+12288,13
*         LH    13,D0004
*         BC    15,E0004
*B0004    DC    Y%A0004+4096¤
*C0004    DC    Y%A0004+8192¤
*D0004    DC    Y%A0004+12288¤
*E0004    EQU   *
```

(BSREG with label, core size specified. Registers 10-13 were assumed.)

```
          BSREG K#12,R1#9,R2#15
*         BASR  9,0
*         USING *,9
*A0001    EQU   *
*         USING A0001+4096,15
*         LH    15,B0001
*         USING A0001+8192,12
*         LH    12,C0001
*         BC    15,E0001
*B0001    DC    Y%A0001+4096¤
*C0001    DC    Y%A0001+8192¤
*E0001    EQU   *
```

(BSREG without label, core size of 12 requested, but only 2 registers supplied. Register 12 is assumed for R3.)

```
          BSREG K#8,R1#13,R2#14
*         BASR  13,0
*         USING *,13
*A0002    EQU   *
*         USING A0002+4096,14
*         LH    14,B0002
*         BC    15,E0002
*B0002    DC    Y%A0002+4096¤
*E0002    EQU   *
```

(BSREG without label, core size of 8 requested, both necessary registers supplied.)

(Generation suppressed due to incorrect core specification.)

```
 TEST6    BSREG K#15
*         MNOTE INCORRECT SPECIFICATION OF CORE SIZE AS 15
```

Exhibit 3

S/360 AND 1401 7 AND 9 TRACK TAPE CHARACTERISTICS

The following is a field contribution. It has been helpful in one installation as an aid in converting from 7 to 9 track tape and selecting the S/360 Hex Code. See the chart contained in Exhibit 1.

S/360-44 FORTRAN I/O DEVICES NOT CLOSED CORRECTION

Installation Newsletter 68-13 contained, on page 6, an article on the above subject. The following errors should be corrected:

1. In the first line of the second paragraph, change WZ to W2.

2. In the suggested fix change WZ to W2 in the first two lines.

3. In the suggested fix (last line), change BOCO8215 to BOCO8005.

The fix is an S/360-44 PS V-3 Assembler subroutine for execution under FORTRAN. The fix follows the label IOTEST1 at BOCO6360 and is for the purpose of proper closing of I/O devices by BOAFIOCS.

1800 USER INFORMATION IN 1130 USER'S GUIDE

The 1130 User's Guide (C20-1690), just released, contains a wealth of material for both scientific and commercial 1130 users. Despite its orientation toward the 1130, though, many of the topics covered will be of interest to 1800 users as well. In fact, because of the similarities of the hardware, the I/O devices, and the programming systems, it is possible to read some sections as if they were written for the 1800 rather than the 1130; other sections require some knowledge of the differences between the two systems.

Here are some of the key topics which can be used by 1800 users with little or no change or interpretation: (items marked with an asterisk are generally applicable, except for a few obvious 1130/1800 differences)

Section	Title
15.10	Data - on disk or cards?
15.20	How to safeguard your disk disk data files
20.20	Forms design
20.30	Card design
20.40	Design of disk data files
30.00	Testing
60.10*	Disk storage layout
60.20*	Increasing the amount of space available to the user
60.30*	Disk Utility Program
65.10*	The logical layout of core storage
70.10	Arithmetic considerations
70.40	Character handling techniques
70.50	FORTRAN core savings tips
80.00	Use of the disk for data storage
90-00*	Improving your system -- performance

S/360 AND 1401 7 AND 9 TRACK TAPE CHARACTERISTICS

1401 Mode on a S/360 requires 6 bits per character.

S/360 Mode requires 8 bits per character.

MODE	# OF TRACKS	# OF BITS	# OF TAPE FRAMES	DENSITY	S/360 HEX	DESCRIPTION	PARITY	FEATURE
1401	7	6	1	800	A8	TRANSLATE	EVEN	N/A
1401	7	6	1	556	68	TRANSLATE	EVEN	N/A
1401	9	8 *	1	800	C0	S/360	ODD	SINGLE DENSITY
1401	9	8 *	1	800	C8	S/360	ODD	DUAL DENSITY
1401	9	8 *	1	1600	C0	S/360	ODD	EITHER WAY
S/360	7	8	1 1/3 **	800	90	DATA CONVERT	ODD	N/A
S/360	9	8	1	800	C0	S/360	ODD	SINGLE DENSITY
S/360	9	8	1	800	C8	S/360	ODD	DUAL DENSITY
S/360	9	8	1	1600	C0	S/360	ODD	EITHER WAY

* Two useless bits of the 8 are dropped when read into core. Dummy bits are added when written.

** For example: Data conversion always requires 4 characters on tape for every 3 in core —

DATA CONVERT

IN CORE			
BYTE #	A	B	C
BITS	1	1	1
	2	2	2
	3	3	3
	4	4	4
	5	5	5
	6	6	6
	7	7	7
	8	8	8
# OF BITS	8	8	8 = 24

ON TAPE				
FRAME #	I	II	III	IV
BITS	A1	A7	B5	C3
	A2	A8	B6	C4
	A3	B1	B7	C5
	A4	B2	B8	C6
	A5	B3	C1	C7
	A6	B4	C2	C8
# OF BITS	6	6	6	6 = 24

Exhibit 1

NEW TYPE III PROGRAMS

The following are the abstracts of programs which have been recently made available from the Type III library.

The program, along with its complete abstract, will be incorporated in subsequent issues of the Catalog of Programs.

Programs may be obtained by submitting a properly completed "General Program Request Card" (Form Number 120-1145-1) to the Program Information Department, 40 Saw Mill River Road, Hawthorne, New York, 10532.

These programs and their related documentation are distributed by IBM in the author's original form and have not been subjected to any formal testing.

Any discussions of Type III programs with customers must emphasize the following points:

1. Type III (IBM employee-contributed) programs are provided by the IBM Corporation as part of its service to customers.

2. Type III programs have not been subjected to any formal test.

3. Recipients of Type III programs are expected to make the final evaluation as to the usefulness of the programs in their own environment.

4. There is no committed maintenance for Type III programs. However, any changes the author chooses to make will be announced in subsequent issues of the Catalog of Programs.

5. IBM makes no warranty, expressed or implied, as to the documentation, function, or performance of Type III programs.

---Revision---

TRANSPORTATION PROBLEM.

New documentation and machine readable material.

Ordering Procedure: Order File Number 360D-15.2.002.

The program material can be obtained in card form or on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

If not specified, a 9 track DTR will be provided. No tape submittal is required. The DTR will be provided by the Library.

---Revision---

STUDENT SCHEDULING SYSTEM/360, DOS VERSION. An errata sheet has been added to the progrm documentation - instructions for overcoming problems associated with the assembly under DOS Release 15 and subsequent DOS releases.

Ordering Procedure: Order File Number 360D-15.0.001.

Order documentation only.

---Revision---

1130 BILL OF MATERIAL MAINTENANCE AND RETRIEVAL SYSTEM.

Revised documentation and machine readable material.

Ordering Procedure: Order File Number 1130-23.3.001.

Distribution will be in card form only.

--Revision--

1800 STORAGE CRT DISPLAY SYSTEM.

Revised program documentation and machine readable material.

Ordering Procedure: Order File Number 1800-03.4.006.

Distribution will be in card form only.

COS-VERSION 5, LEVEL 1 COMPATIBILITY OPERATING SYSTEM 360D-05.1.005. (Replaces Version 5, Level 0).

Documentation and machine readable changes.

Significant Differences from Earlier Version of "COS"

This release of the Compatibility Operating System incorporates many corrections to the existing version as well as additional support and performance enhancements in certain areas. Users of earlier versions of the Compatibility Systems should pay particular attention to the following items which are implemented under this release in a different manner.

1. Includes support for the simulation of the 1400 card reader, card punch, and printer on 2314 disk extents.

2. Includes support for the simulation of up to five (5) 1311's on 2314 disk storage. As many as four (4) 1311's may be simulated on a single 2314 module.

3. Includes support for the simulation of a 1301 module on two and one-half (2 1/2) modules of 2314 disk storage.

4. Includes support for the simulation of one or two modules of 1405 disk storage on 2314 disk storage. Each 1405 module on separate 2314 modules.

5. Includes support for the simulation of disk file scan operations utilizing the File Scan Feature (#4385). When the number of sectors to be scanned exceeds 20, a performance improvement will be realized over previous versions.

6. Includes support for selective 1405 disk verification on a drive for drive basis consistent with 1311 support.

7. Includes support for an additional (optional) 1400 control card parameter enabling the user to specify at execution time the instruction address of the normal 1400 end-of-job Halt instruction. Upon encountering this halt, control will be automatically transferred to DOS Job Control.

8. Includes an improved Program Mode Switch Supervisor Call interrupt handler.

9. Includes support for the simulation of 1401/1440/1460 Write and Space Suppresses operation and 1401/1460 Write Word-Mark operation.

10. Includes an improved simulation technique for the simulation of 1400 carriage control operations.

11. Includes performance enhancements for the simulation of 1402 card reader stacker select operations.

12. Includes support for the simulation of 1401/1460 Read Column Binary operations with optional translation of the binary data in the normal card input area.

Ordering Procedure: Order File Number 360D-05.1.005.

The program material can be obtained in card form or on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

If not specified, a 9 track DTR will be provided. No tape submittal is required. The DTR will be provided by the Library.

DEMONSTRATION INFORMATION SYSTEM. This is designed to demonstrate the functions and capabilities of IBM terminals operating on line with System/360 under the control of DOS/QTAM in a multi-programming environment.

Separate applications are provided for each of the terminal devices supported as follows: (1) 1030 Receiving and Production Reporting; (2) 1050 Inquiry on stock status and accounts receivable, Order Entry, Shipping Confirmation; (3) 2260 sales analysis; (4) 2740 conversational entry in purchasing environment; (5) 2780 order entry and invoicing (using BTAM).

DIS requires a S/360 Model 30 64K. (See the following configuration list for machines and features required.) The source language used for all applications is 360 Basic Assembler Language.

See the following lists.

Ordering Procedure: Order File Number 360D-30.1.001.

The program material can be obtained in card form or on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

If not specified, a 9 track DTR will be provided. No tape submittal is required. The DTR will be provided by the Library.

1050 Data Communications System

1051-1		Control Unit
	4408	1st Printer Attachment
	4410	1st Punch Attachment
	4411	1st Reader Attachment
	6384	2nd Reader Attachment
	4770	Keyboard Request
	4795	Line Correction
	5465	Open Line Detection
	4605	Home Component Recognition
	9114	Data Set Attachment-WE 103A
1052-1		Printer-Keyboard
	5465	Open Line Detection
1054-1		Paper Tape Reader
1055-1		Paper Tape Punch
1056-1		Card Reader

2260 Visual Display

2260-2		Display Station
	4760	Alphameric Keyboard

T-P TERMINALS SUPPORTED BY DIS

1030 Data Collection System

1031-A4		Input Station
or		
1031-A5		
	1279	1033 Printer Attachment
1033-1		Printer
	9575	Standard Printing Element

2260 Visual Display - Local

2260-2 Display Station
- 4766 Keyboard, Alphameric

1053-4 Printer
- 1006 Accelerated Carrier Return
- 9104 Character Spacing 10/Inch
- 9354 Line Space 6 LPI Holes 8
- 9435 Line Feeding, 6 Lines/Inch
- 9509 Pin Feed Plater, Regular
- 9571 Dual Case PR EL-S/360

2848-2 Display Control
- 3356 Display Adapter for Model 2
- 3858 Expansion Unit-Models 1, 2
- 4787 Line Addressing
- 5340 Non-Destructive Cursor
- 5341 Non-Destructive Cursor Adap.
- 7927 1053 Adapter for Model 1, 2
- 9011 Channel Adapter

2848-1 Display Control
- 3356 Display Adapter for Model 2
- 4787 Line Addressing
- 9012 Data Set Adapter - 1200 BPS

2740 Communications Terminal

2740-1 Communications Terminal
- 3255 Dial-Up
- 6114 Record Checking
- 8028 Transmit Control
- 9114 Data Set Attachment-WE 103A

2780 Data Transmission Terminal

2780-2 Data Transmission Terminal
- 1340 Automatic Answering
- 1350 Automatic Turnaround
- 5010 Multiple Record Transmission
- 5800 Printer Horizontal Format Control
- 5820 Print Line for 120-Characters
- 5821 Print Line for 144-Characters
- 6400 Selective Character Set
- 909847 Char. Set for 6 Bit Trans.
- 9110 Data Set Att-WE 201A3
- 9710 For Commun. W. Processor Sys.
- 9760 Transmission Code-6 Bit Trans.

APL/360 is a conversational time-sharing system based on a mathematical programming language first developed by K. E. Iverson. The language is concise and has a simple syntax. It has a large set of primitive operations which work directly on arrays. The implementation provides a simple immediate-execution mode and a convenient program definition facility. It has fast response, and uses succint diagnostic messages. It provides the ability to save work between sessions, to create programming packages, and to exchange programs and data between users. Uses of the system include mathematical and statistical calculation, symbol manipulation and general data processing. It has been used extensively in computer-related instruction, and in the design of hardware and software.

Minimum configuration: 256K storage; universal instruction set; a 2314 or three 2311's; one 2702 or 2703; and 2741 or 1050 terminals. Additional storage, disks, and transmission controls are supported. A typical Model 50 configuration supports 60 terminals. The system is written in Assembly language, incorporates a modified DOSIII, and allows standard DOS background operation.

Ordering Procedure: Order File Number

360D-03.3.007.

The program material can be obtained on magnetic tapes as follows:

1. For 2311 users - 2 magnetic tapes are required.

2. For 2314 users - 3 magnetic tapes are required.

Submit the appropriate number of reels of 2400 foot magnetic tape. Specify whether 7 or 9 track recording is required. If not specified, 9 track recording will be provided. The required tapes may be ordered from IBM or supplied with the request for the program.

Customers wishing to use the 2311 system with Teleprocessing components of DOS must order the DOS Teleprocessing components separately.

The following SRL publications would be helpful to users of this program:

1) C24-5022 - DOS Operation Guide; 2) C24-5033 - DOS System Generation and Maintenance; 3) C24-5036-System Control and System Service Program.

The above SRL publications should be ordered from the IBM Distribution Center, Mechanicsburg and will not be distributed with this program or available from PID.

ELECTRONIC SYSTEM CONVERSION OF AUTOCODER PROGRAMS. This program translates 1401, 1440, or 1460 Autocoder source programs to S/360 DOS ALC or S/360 Mod 20 DPS or TPS source programs and is not a demonstration program. The program is submitted to help users get into 360 mode exclusively.

Any system using DOS with 32K and the Packed Decimal feature is sufficient to execute this program.

This program is written in 360 DOS Assembly Language.

Ordering Procedure: Order File Number 360D-12.2.006.

The program material can be obtained in card form or on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

If not specified, a 9 track DTR will be provided. No tape submittal is required. The DTR will be provided by the Library.

ENGINEERING SPOOLING PROGRAM. The Model 44 Engineering Spooling Program (ESP) modifies the S/360 Model 44 Programming System to permit I/O requests for the card reader printer, and punch to be intercepted and satisfied by high speed core-to-core transfers. Jobs which involve I/O operations at the read/write level can operate without change. This program attempts to read cards ahead of their actual use, storing them in internal core buffers and on a disk. Printer and punch records are also similarly queued. Up to 200 cylinders of a 2311 or SDSD can be utilized for reader, printer, and punch image queuing. The ESP Package is written in 44PS Assembler Language and requires from 25K to 41K, depending on User-Specified buffer sizes. A 128K or 256K CPU is required to satisfy the combined requirements of 44PS and ESP. Unique device addresses for reader, printer, and punch are required. Users with a card-disk 44 PS system should order the program in card form. Users with a tape-disk system should order the program in tape form. About 4000 source statements are included in the necessary distributed material.

Ordering Procedure: Order File Number 360D-03.4.023.

The program material can be obtained on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

If not specified, a 9 track DTR will be provided. No tape submittal is required. The DTR will be provided by the Library.

1130 FORTRAN DOCUMENTATION AND FLOWCHARTING PROGRAMS. Program documentation and flowcharting is often a problem in smaller data processing installations. Because of the desirability of having good documentation, these programs were written. Input to these programs is in the form of FORTRAN source statements and comments cards. Output from these programs is a complete flowchart of the FORTRAN program. These programs are a modification of 360D-00.2.001, BPS/BOS FORTRAN Flowchart Program, for the use on an 8K, 1 disk, 1130 system operating under control of the 1130 Disk Monitor System. (Version 1 and Version II).

Ordering Procedure: Order File Number 1130-00.2.001.

Distribution will be in card form only.

A CARD UTILITY PROGRAM FOR 1800/ TSX. The Card Utility Program for 1800/TSX provides a series of functions to list, reproduce, sequence number and convert Hollerith/EBCDIC data codes. The program, written in Assembler Language, runs as a non-process core-load under TSX Version 2 or Version 3 (1800-OS-001). The 1800 must have a 1443 Line Printer in addition to the minimum requirements for TSX.

Ordering Procedure: Order File Number 1800-00.1.001.

Distribution will be in card form only.

IBM Installation Newsletter

October 18, 1968

Issue No. 68-20

CONTENTS

(G means Green Section)

Distribution: Branch Offices - DP Management, Salesmen, Systems Engineers, FE Managers. Regions, Districts, Education Centers, Field Systems Centers, Federal Systems Centers, FE Area Offices, DPD HQ FED HQ, WTC.

*Requires Immediate Attention

For IBM Internal Use Only

DPD HQ, White Plains, N. Y.

OS/360 TIMER ACCURACY

The use of the interval timer under OS/360 to measure very small intervals should be undertaken only after careful consideration of factors which may affect its accuracy. These factors are:

1. The theoretical resolution of the timer is not exactly 26 microseconds as stated in the Supervisor and Data Management Macro Instructions SRL, rather it is 26.041666 microseconds. This amounts to a 0.16% variance.

2. The timer is impulsed at the frequency of the alternating current used to power the machine (either 50 or 60 cps). Line frequency can fluctuate by several cps. This will cause the timer to exhibit equivalent fluctuations.

3. Under certain contention situations, a memory cycle may not be taken, with the result that the timer may not be decremented.

The occurrence and the effect of factors 2 and 3 cannot be predicted. Therefore, use of the timer to measure very small intervals may not result in the desired accuracy.

OS/360 RELEASE 13 AND 14 UTILITY RECORD OVERFLOW SOLUTION

The following is a field contribution which has not been submitted to any formal IBM test. Potential users should evaluate its usefulness in their own environment prior to implementation including maintenance and the current availability of OS/360 Release 15/16.

Versions of the independent utility program IBCDMPRS included in OS/360 Releases 13 and 14 do not correctly support record overflow on the 2841 control unit. This occurred in an installation when a data set using record overflow would not restore correctly on a 2311. There was no indication of this until the data set was used.

The problem can be overcome by changing to another version of IBCDMPRS. A more serious problem has occurred if it is necessary to recover information dumped using one of the above versions of IBCDMPRS.

Exhibit 1 contains an example of a subroutine approach designed to recover a dump of a 2311 pack using the Release 13 version of IBCDMPRS. Input consists of the dump tape which is to be restored, and output is a tape that can be used with the Release 13 IBCDMPRS to correctly restore the pack and set the appropriate flag bits which indicate record overflow.

OS/360 RELEASE 13 AND 14 UTILITY RECORD OVERFLOW SOLUTION

```
         PRINT NOGEN
A        START 0
         SAVE (14,12)
         BALR 5,0
         USING *,5
         ST    13,SAVE+4
         LR    12,13
         LA    13,SAVE
         ST    13,8(12)
*           USUAL LINKAGE CONVENTIONS HAVE BEEN MADE
         OPEN (DATAIN,(INPUT))
         OPEN  (OUTPUT,(OUTPUT))
         GET DATAIN,CHANPG
*   THIS GET OBTAINS THE PARM RECORD FOR DUMP-RESTORE
         MVC OUTPUT+82(4),DATAIN+82   MAKE WRITE LENGTH EQ READ LENGTH
         PUT   OUTPUT,CHANPG
         WTO 'PARM RECORD WRITTEN'
STEPONE  GET DATAIN,CHANPG
         MVC CHANLEN(4),DATAIN+82
         GET   DATAIN,DATA
         MVC DATALEN(4),DATAIN+82
*  THE CHAN PROG AND TRK IMMAGE ARE NOW IN CORE
         CLC DATA+14(2),OE29      IS THE TRK COMPLETELY FULL
         BNE   WRITE
         CLI CHANPG+56,X'ID'
         BE    RECOVER
         WTO   'OTHER THAN ID OPCODE'
         BC 15,EOJ
SND1     WTO 'SYNAD ON READ'
         BR 14
SND2     WTO 'SYNAD ON WRITE'
         BR 14
RECOVER  MVI CHANPG+56,X'01'
*    THE CHAN PROG TO RESTORE THIS TRK HAS BEEN CHANGED TO
*          INDICATE WRITE SPECIAL COUNT,KEY,DATA VICE WRITE
WRITE    MVC OUTPUT+82(4),CHANLEN
         PUT   OUTPUT,CHANPG
        MVC OUTPUT+82(4),DATALEN
         PUT   OUTPUT,DATA
         BC    15,STEPONE
EOJ      CLOSE (OUTPUT,DISP)
         CLOSE (DATAIN,DISP)
         WTO 'DATA SETS CLOSED'
         L     13,SAVE+4
        RETURN (14,12)
DATAIN   DCB BUFNO=1,BLKSIZE=4000,DDNAME=TAPEIN,MACRF=(GM),DEVD=TA,      XXX
               DSORG=PS,EODAD=EOJ,EROPT=ACC,SYNAD=SND1
               *,***  IH8063  DDNAME SHORT-PADDED TO 8 CHAR
OUTPUT   DCB BUFNO=1,BLKSIZE=4000,DDNAME=TAPEOUT,MACRF=(PM),DEVD=TA,     XXX
               DSORG=PS,SYNAD=SND2
CHANLEN  DS   F
DATALEN  DS   F
SAVE     DS   18F
OE29     DC   X'0E29'
BLANK    DC   CL1' '
         LTORG
CHANPG   DS   CL500
DATA     DS   CL4000
         END  A
```

EXHIBIT I

DOS/360 BASIC PL/I AND COBOL SUPPORT

The following is a field contribution. Exhibit 1 contains a chart on the Direct Access Capability for PL/I and COBOL, current as of DOS/360 Release 18. Additional information on the new capabilities for Basic PL/I is contained in Program Announcement Letter P68-114, dated August 15, 1968.

DOS/360 BASIC PL/I AND COBOL SUPPORT

DIRECT ACCESS CAPABILITY

FEATURES	PL/I	COBOL
	GENERAL	
DEVICES	2311, 2314, 2321	2311, 2314, 2321
ORGANIZATION	IS, REGIONAL(1), REGIONAL(3)	S, IS, DIRECT, 5
VERIFICATION	NOT ASSUMED, SPECIFY BY KEY WORD	ASSUMED, MAY CHANGE BY SEPARATE ROUTINE MODIFYING BIT IN DTF TABLE
	INDEX SEQUENTIAL	
MASTER INDEX	YES	NO- MUST CHANGE BIT IN DTF TABLE THRU SEPARATE ROUTINE
INDEX LOC	MAY SPECIFY DIFFERENT DEVICE TYPE, EX PRIME DATA ON 2321, INDEX ON 2311	NO- MUST CHANGE BIT IN DTF TABLE THRU SEPARATE ROUTINE
IN CORE CYL INDEX	YES	NO SUPPORT
FULL TRACK ADD	YES	NO SUPPORT
OVFLO TRACKS	USER MAY SPECIFY	STANDARD 20% ASSUMED, CAN Be Changed Only By Modifying BIT IN DTF TABLE THRU SEPARATE ROUTINE
SETL/ESETL	YES	Limited-SETL can be done only once at open time and ESETL at close time.
	DIRECT ORGANIZATION	
	REGIONAL(1)- RECORDS LOADED, RETRIEVED BY RELATIVE RECORD NO. REGIONAL (3)-RECORDS LOADED, RETRIEVED BY RELATIVE TRACK + RECORD KEY	NOTHING EQUIVALENT DIRECT-RECORDS CREATED AND RETRIEVED BY SYMBOLIC KEY AND ACTUAL KEY (8 BYTE TRACK ADDRESS)

EXHIBIT 1

DOS/360 BILL OF MATERIAL PROCESSOR LINKAGE EDITOR TECHNIQUE

The following is a field contribution which has not been submitted to any IBM test. Potential users should evaluate its usefulness in their own environment prior to implementation.

When cataloging a load module to the Core Image Library with the same phase name as one already in the library, only the latest phase is accessable to DOS. This applies to Bill of Material Processor phases for I/O Open/Close, I/O Process, and Dearux when several programs of a system use these same modules. Three approaches may be used to link edit these program phases to the CIL to assure proper execution of the desired program.

First, the phase names may be changed, along with the corresponding Fetch Operand, so that a unique phase name exists for each program's I/O Open/Close, I/O Process, and Dearux overlays. This wastes space in the CIL, since each similar phase consists of the same code, except for address constants. Changing and keeping the various fetch operands and phase names correct is a tedious job.

Second, the three overlay phases may be link-edited at the same location in high core, instead of beginning at XZ$OVLY, since the location of XZ$OVLY is variable from program to program. This technique results in each of the three phases in the CIL only once and under the same name at the same executable core location. Of course, they must be high enough in core so that a future program's root phase that uses them will not be overlaid.

Third, and the suggested tehcnique is to link edit the three overlay phases immediately following the label area, with the main or root phase following the end of the largest of the three phases.

This technique makes use of 2 desirable properties of the DOS Linkage Editor. Only the necessary number of bytes for a C. I. L. phase are brought into core, not necessarily a full block (1728 or 1688 bytes). This results in the shortest overlay possible.

Also, core is not cleared from the end of the fetched phase to high core. If it were, the root phase would be cleared.

The phase cards generated by BMP for the overlay point at XZ$OVLY must be discarded from the I/O Process and Open/Close modules, and the similar one for Dearux may be deleted by doing an RSERV, discarding, and re-cataloging the module.

The I/O Root module generated by BMP is core location dependent when used from one program to another, so it is placed first in the root phase. See Exhibit 1 for a Sample Linkage Editor run.

DOS/360 BILL OF MATERIAL PROCESSOR LINKAGE EDITOR TECHNIQUE

```
// JOB MAC CATALOG PAC1010D                                    14.44.28
// OPTION CATAL
   ACTION CLEAR
   ACTION MAP
   PHASE DEARUX,*,NOAUTO
   INCLUDE
   PHASE IOD1020D,DEARUX,NOAUTO
   INCLUDE IOD1020D
   PHASE IOD1030D,DEARUX,NOAUTO
   INCLUDE IOD1030D
   PHASE PAC1010D,*
   INCLUDE IOD1010D
   INCLUDE IOC1010D
   INCLUDE
// LBLTYP NSD(3)
// EXEC LNKEDT
```

```
JOB  MAC      08/28/68  DISK LINKAGE EDITOR DIAGNOSTIC OF INPUT

ACTION TAKEN    MAP  CLEAR
 LIST      PHASE DEARUX,*,NOAUTO
 LIST      PHASE IOD1020D,DEARUX,NOAUTO
 LIST      INCLUDE IOD1020D
 LIST      PHASE IOD1030D,DEARUX,NOAUTO
 LIST      INCLUDE IOD1030D
 LIST      PHASE PAC1010D,*
 LIST      INCLUDE IOD1010D
 LIST      INCLUDE IOC1010D
 LIST    INCLUDE IHD02100                                  PAC10001
 LIST    INCLUDE IHD03700                                  PAC10002
 LIST    AUTOLINK   GETDATE
 LIST    AUTOLINK   IJCFZ1I0
 LIST    AUTOLINK   IJCFZ0I4
 LIST    AUTOLINK   IJDFAP1Z
 LIST    AUTOLINK   IJGFOZZZ
 LIST    ENTRY
```

EXHIBIT I

DOS/360 BILL OF MATERIAL PROCESSOR LINKAGE EDITOR TECHNIQUE

08/28/68	PHASE	XFR-AD	LOCORE	HICORE	DSK-AD	ESD TYPE	LABEL	LOADED	REL-ER
	DEARUX	002890	002890	002ACB	3F 01 2	CSECT	DEARUX	002890	002890
	IOD1020D	002890	002890	0030CE	3F 02 1	CSECT	IOD1020D	002890	002890
						ENTRY	YA$RT	002890	
	IOD1030D	002890	002890	00310F	3F 03 1	CSECT	IOD1030D	002890	002890
	PAC1010D	007304	003110	008661	3F 04 1	CSECT	IOD1010D	003110	003110
						ENTRY	XA$RET1	00317A	
						ENTRY	XL$COMM	003238	
						ENTRY	BM$PIO	003110	
						ENTRY	MS$GRT	003190	
						* ENTRY	XZ$OVLY	00552E	
						CSECT	IOC1010D	005EE8	005EE8
						CSECT	BMPOPEN	0061E0	005EE8
						CSECT	BMPCLOSE	006250	005EE8
						CSECT	BMPCALL	0062B0	005EE8
						CSECT	BMPGET	006338	005EE8
						CSECT	BMPPUT	006360	005EE8
						CSECT	BMPSTKY	006388	005EE8
						CSECT	BMPSTDA	0063B0	005EE8
						CSECT	BMPEOF	0063D8	005EE8
						CSECT	IHD02100	006410	006410
						ENTRY	IHD02101	006410	
						CSECT	IHD03700	0064C8	0064C8
						* ENTRY	IHD03701	0064C8	
						* ENTRY	IHD03702	0064D8	
						ENTRY	IHD03704	00665C	
						CSECT	PAC1010D	006768	006768
						CSECT	IJGFZIIO	0081D8	0081D8
						CSECT	IJGF0ZZZ	008460	008460
						CSECT	GETDATE	0081C0	0081C0
						* ENTRY	GETDAT	0081C0	
						CSECT	IJDFAPIZ	008348	008348
						* ENTRY	IJDFAZIZ	008348	
						CSECT	IJGFZOI4	008260	008260

EXHIBIT I (Cont'd)

S/360-20 TRANSLATING PAPER TAPE INPUT TO EBCDIC

The following is a field contribution. RPQ's are required which must be approved by your Regional Special Equipment Engineering department prior to making any customer commitment. The usefulness of the approach used in this contribution should be evaluated in the potential users own environment prior to implementation.

This article includes an example which illustrates the use of a translate table in translating 8-track paper tape to EBCDIC using the 3903-4 Paper Tape Reader (RPQ Y95017). A Paper Tape Reader Adapter (RPQ Y91034) is also required on the 2020.

The 3903 can read 5, 7 or 8 track paper tape and can be interfaced with a S/360-20 as an input device. In the following example, 8-track paper tape code is translated into EBCDIC.

The 127 byte table shown in Exhibit 1 must be generated to contain the characters of the EBCDIC Code in order by the binary sequence of the characters of 8-track paper tape code.

NOTES:

1. The unused bytes within the table could be filled with some character (i. e., FF) which could be used as a check.

2. Additional characters could be added to the table using the following scheme: tape tracks, 8, 7, 6, 5, 4, 3, 2, and 1 correspond to EBCDIC bits 0, 1, 2, 3, 4, 5, 6, and 7 respectively.

Specifically, the 3903 Paper Tape Reader can utilize this translate table as shown in Exhibit 2.

S/360-25 1440 COMPATIBILITY

To run 1440 programs on a Model 25, a 1442 attached to a selector channel is necessary at the present time. To run these same programs on a Model 25 with only a 2540, the card and printer routines need to be changed to reference a 1402. The program then becomes essentially a 1401 program and 1401 Compatibility would be used. 1442 emulation on a 2540 was announced in a DP Product Announcement Letter, #268-58 on July 8. It will be available January 1, 1969.

S/360-20 TRANSLATING PAPER TAPE INPUT TO EBCDIC

Address of Translate Table → xxoo

xx10		1	2		4			7	8			#			
xx20	↻			3		5	6			9			@		
xx30	0			T		V	W			Z			%		
xx40		/	S		U			X	Y			,			
xx50	-			L		N	O			R			*		
xx60		J	K		M			P	Q			$			
xx70		A	B		D			G	H						
	&			C		E	F			I					

XX0F

XX7F

EXHIBIT I

S/360-20 TRANSLATING PAPER TAPE INPUT TO EBCDIC

Location into which characters are read ↓ Number of characters to be read ↓

```
XIO  O(X'UF',9),30
```

where UF = 02 for regular read where the number of characters to be read is indicated by the second operand (30 in this example)

where UF = 04 for modified read where the number of characters read is terminated by an End of Record character or length count = 1.

Number of characters to be transmitted ↓ Location of characters ↓

```
TR   O(30,9), TABL  ←— Label of translate table
-
-
TABL DC   X'FFF1F2FFF4FFFFF7F8FFFF7BFFFFFF
     DC   X'FF4OFF  ,  ETC.
```

NOTES:

1. With the DELETE switch on, fields with all eight tracks punched are bypassed. This may be used to correct errors in the tape.

2. The E.O.R. punch is an eight punch; this character is read but not transmitted to the CPU. Therefore, it isn't part of the field, as far as length count is concerned when you use modified read. The number 8 E.O.R. switch must be on to take advantage of modified read.

3. The residual count (difference between length count and E.O.R.) is located in X'A4', X'A5'.

EXHIBIT 2

INFORMATION ABOUT THE NEWSLETTER

EVERY SALESMAN AND SYSTEMS ENGINEER SHOULD RECEIVE ONE COPY OF THE NEWSLETTER. Each FE Branch Manager should receive five copies for distribution to Customer Engineers.

The IBM Installation Newsletter is distributed only to IBM locations and is not available to customers. The Newsletter is separated into two parts. The first part contains information which is intended only for IBM personnel and the pages are labeled For IBM Internal Use Only. THESE PAGES, OR REPRODUCTIONS OF THEM, ARE NOT TO BE GIVEN TO CUSTOMERS. However, using your own good judgement, you may discuss pertinent information from these pages with a customer. The second part contains information which may be REPRODUCED by the Branch Office at their discretion and given to customers. These pages are not labeled with any restrictive classification. Note that when any abstracts for Type III programs are given to the customer, he must be informed that the disclaimer for Type III programs as contained in the Newsletter, applies. YOUR COOPERATION IN PROPERLY HANDLING THE CONTENT OF THE IBM INSTALLATION NEWSLETTER IS NECESSARY FOR ITS CONTINUED PUBLICATION.

Changes to the quantity of the Newsletter shipped are to be made by the Administration Manager. (NOTE THAT EXTRA COPIES OF THE NEWSLETTER ARE NOT TO BE ORDERED FOR DISTRIBUTION OF INFORMATION TO CUSTOMERS.) He should update and send the DP Administration Managers' Communication Control card (form number N44-780) to DP Document Control, DPD HQ.

Direct inquiries concerning the contents of this Newsletter to:

W. P. Gaffigan, Editor
IBM Installation Newsletter
Installation Planning-#713
DPD HQ
112 East Post Road
White Plains, New York 10601
Telephone 914 WH9-1900 Extension 4781

NEW PUBLICATIONS ANNOUNCED BY PRL'S

Information is extracted and condensed from the weekly PRL's (Publications Release Letters) to insure that all Salesmen and Systems Engineers are aware of new or revised Marketing Publications. Normally, each issue of the Newsletter will contain information extracted and condensed from two PRL's, one following the other. The information will be placed in the Newsletter in its original sequence with no rearrangement of form numbers or titles. It is not intended to replace existing information and distribution sources. You should be certain that you are aware of these sources.

NOTE:

Do not use the condensed Newsletter information to order publications. Most but not all (for example, some films and slide sets) of the items are available from Mechanicsburg. To receive the items you need without delay, prior to ordering check the REQ (Requisition From) code in the "Accumulative Supplement to IBM Publications Current Price List (Z20-0100)" which is attached to the weekly Publications Release Letter (PRL) and distributed to your Administration Manager. For an explanation of the REQ codes and other publications information see the first few pages of the "IBM Publications Current Price List (Z20-0100)".

The Publications Requisition, form number M02-0618 is used for ordering items from the IBM Distribution Center, Mechanicsburg, Pa.

(IBM World Trade Corporation has its own distribution center and procedures.)

PRL #38 September 20, 1968

FORM NO.	TITLE	DISPOSITION
N20-0013-15	Catalog of Programs for 1240-1401-1420 and 1460 DPS - September 1968 Supplement to C20-1601-9	NEW
N20-0015-15	Catalog of Programs for IBM 1620 and 1710 DPS September, 1968 Supplement to C20-1603-9	NEW
N20-0030-22	Catalog of Programs for IBM S/360 - September 1968 Supplement to C20-1619-5	NEW
N20-0031-18	Catalog of Programs for 1130 Computer System and 1800 Data Acquistion System - September 1968 Supplement to C20-1630-4	NEW
N20-0360.50	IBM S/360 Newsletter Re: A22-6822-11	Scrap N20-0360.49
N20-1852-2	Catalog of Programs for IBM S/360 Model 20 September 1968 Supplement to C20-1691-1	NEW
	MARKETING PUBLICATIONS	
270-0092-0	Instructions, Jacketed Page Holder	NEW
540-0504-0	Certificate of Completion	NEW - Scrap 540-0501
543-0062-0	Model "D" Short Course Booklet	NEW
543-0511-0	RR MT/ST Instruction Manual	NEW
543-0512-0	RR Mt/ST Instruction Card	NEW
570-0295-3	One-Half Inch Mag. Tape Spec. Booklet	Use 570-0295-2
570-0417-0	DP Accessories Folder - System/360 Mod 25	NEW
570-0418-0	DP Accessories Folder - System/360 Mod 30	NEW
570-0419-0	DP Accessories Folder - System/360 Mod 40	NEW
570-0420-0	DP Accessories Sheet - General Purpose Cards	NEW
570-0421-0	DP Accessories Sheet - Ind Application Cards	NEW
570-0423-0	DP Accessories Sheet - DP Ribbons	NEW
570-0424-0	DP Accessories Sheet - Paper Forms	NEW
570-0425-0	DP Accessories Sheet - Magnetic Tape	NEW
H20-0255-2	IBM S/360 Wholesale IMPACT Program Library (360A-DW-05X Version 2, and 360V-DW-06X) Program Description Manual	Scrap H20-0255-1
H20-0256-2	IBM S/360 Wholesale IMPACT Program Library (360A-DW-05X Version 2, and 360V-DW-06X) Operations Manual	Scrap H20-0256-1
H20-0461-1	IBM S/360 Model 20 Wholesale IMPACT Program Library (360V-DW-06X) Application Description	Scrap H20-0461-0 & N20-1849
H20-0551-0	1450/1259 Deman Deposit Conversion Program (1450-FB-20X) Program Descritpion Manual	NEW
H20-0552-0	1450/1259 Demand Deposit Conversion Program (1450-FB-20X) Operations Manual	NEW
N21-0095	1287 Optical Reader SRL TNL Re: A21-9064-2	NEW
R20-4120-0	IBM S/360 Operating System Utilities for Operators (Learner's Guide Text)	NEW
R20-4124-0	Intermediate Course for Application Programming (ICAP)	NEW
R20-9072-2	IBM 82 Sorter Operation - Programmed Instruction	Scrap R20-9072-1
R20-9268-0	Intermediate Course for Application Program (ICAP) Course Description	NEW
V20-0016-0	IBM Inventory Management Simulator (slides)	Revised Abstract
V20-0101-0	IBM Operating System/360 - Data Management (slides)	Revised Abstract
X20-1770-0	BPS/BOS/DOS/TOS Sort/Merge Reference Card	NEW
Y20-0232	TNL for ALIS (360A-IL-09X) File Maintenance Run Executive Program - System Manual Re: Y20-0181-0	NEW
Y20-0234	TNL for ALIS (360A-IL-09X) File Maintenance Include (R) Routines (Narratives) System Manual Re: Y20-0183-0	NEW
Y20-0244	TNL for ALIS (360A-IL-09X) Output Analysis Program - System Manual Re: Y20-0197-0	NEW
Y20-0250-0	1450/1259 Demand Deposit Conversion Program (1450-FB-20X) System Manual	NEW
Z20-1897-29	RPQ Reference List #29	NEW
Z77-8084-0	Generating a Disk Operating System Version 3	NEW
Z77-8095-0	Modifications to OS/360 to Allow Use of 2301 the Drum with 2361 Large Core Storage	NEW
Z77-8096-0	The Use of the Compatibility Operating System for Upgrading of Card Systems to Disk	NEW

PRL #39 September 27, 1968

N20-1440-52 and N20-7070-34 which are announced in Publications Release Letter #39 are not available for shipment at this time and will be shipped at a later date.

FORM NO.	TITLE	DISPOSITION
N20-1440.52	1440 SRL Newsletter Re: A24-3005-S	Scrap N20-1440.51
N20-7070.34	7070 SRL Newsletter E: A22-6699-4	Scrap N20-7070.33
Z20-1708-2	Bibliography of IBM Classified Marketing Aids	Scrap Z20-1708-1
	MARKETING PUBLICATIONS	
250-0043-0	Fundamentals of Systems Science Education	NEW
320-1906-8	August - 1968 In Brief	Scrap 320-1906-7
520-1585-2	A Profile of IBM	Scrap 520-1585-1
520-1614-0	Information Management IBM S/360	NEW
520-2083-1	With IBM's 2760 Optical Image Unit, computer data entry is as easy as pointing your finger	Use 520-2083-0
520-2115-0	CALL/360: DATATEXT	NEW
542-0047-2	Selectric Instruction Book	Use 542-0047-1
545-0084-0	DE State & Local Government Catalog	NEW
549-0061-0	Composer Ruling Font Brochure	NEW
549-0708-1	MTSC Conversion Scale	Scrap 549-0708-0
A27-3005-2	Changes to 2780 Data Transmission Terminal SRL Comp. Description	Scrap A27-3005-1
C27-6942-0	IBM S/360 Operating System Introduction to Main Storage Hierarchy Support for IBM 2361 Core Storage	NEW
C28-2001-2	IBM S/360 Time Sharing System Command System User's Guide	Scrap C28-2001-1
C28-2003-3	IBM S/360 Time Sharing System Concepts and Facilities	Scrap C28-2003-2
C28-2006-0	IBM S/360 Time Sharing System Time Sharing Support System	NEW
C28-2010-3	IBM S/360 Time Sharing System System Generation and Maintenance	Scrap C28-2010-2
C28-2024-1	IBM S/360 Time Sharing System Manager's and Administrator's Guide	Scrap C28-2024-0
C28-2025-2	IBM S/360 Time Sharing System FORTRAN Programmer's Guide	Scrap C28-2025-1
C28-2032-2	IBM S/360 Time Sharing System Assembler Programmer's Guide	Scrap C28-2032-1
C28-2033-3	IBM S/360 Time Sharing System Operator's Guide	Scrap C28-2033-2
C28-2043-4	IBM S/360 Time Sharing System Addendum	Scrap C28-2043-3
H20-0464-1	IBM S/360 Vehicle Scheduling Program (360A-ST-06X) Application Description	Scrap H20-0464-0
H20-0539-0	Graphic Analysis of Three-Dimensional Data (GATD) Application Description Manual	NEW
H20-0546-0	IBM 1130 Charge Materials Allocation Processor (1130 CMAP) Application Description Manual	NEW
K20-0086-0	Alphameric File Search Using IBM 2260 Display Stations at Security Life and Trust Co.	NEW
N20-1879	TNL for System/360 Model 20 Telephone Revenue Accounting System (360V-SU-11X) Version 2 User's Manual Re: H20-0264-1	NEW
N20-1880	TNL to MARVEL/360 (360A-CO-15X) Program Description Manual Re: H20-0505-0	NEW
R20-4107-0	IBM S/360 Model 44 Data Acquisition Multiprogramming System (DAMPS)	NEW
R20-4121-0	IBM S/360 Operating System Utilities for Operators (Learner's Guide - Illustrations)	NEW
R20-4122-0	IBM S/360 Operating System Operator Training Guide	NEW
R20-9235-0	IBM S/360 Model 44 Data Acquisition Multiprogramming System (DAMPS) Course Description	NEW
R20-9256-0	IBM S/360 Model 20 Installation Programming - Card Learner Paled Format	NEW
R20-9258-0	IBM Punched Card Data Processing Accounts Receivable	NEW
R20-9270-0	IBM Punched Card Data Processing Accounts Payable	NEW
R20-9272-0	IBM Punched Card Data Processing General Ledger Accounting	NEW
R20-9273-0	IBM Punched Card Data Processing Inventory Control	NEW
R20-9274-0	IBM Punched Card Data Processing Order Writing & Billing	NEW

PRL #39 (Continued)

FORM NO.	TITLE	DISPOSITION
R29-0113-1	IBM S/360 PL/I Coding An Introduction to PL/I Supplement	Use R29-0113-0
R29-0245-3	IBM Computing System Fundamentals Text-Unit IV	Use R29-0245-2
V20-0162-0	IBM S/360 Model 25 (slides)	Revised Abstract
V20-0186-0	IBM Tele-processing... an executive introduction (slides)	NEW
V20-4579-0	IBM S/360 (movie)	Revised Abstract
V20-4629-0	Man and Computer..A Perspective (movie)	NEW
Y20-0168-0	IBM S/360 Vehicle Scheduling Program (360A-ST-06X) Schedule Production and Distance Listing System Manual	NEW
Y20-0222-3	Sales Manual Inserts (less programming)	Scrap Y20-0222-2
Y20-0223-3	Sales Manual Inserts (programming only)	Scrap Y20-0223-2
Y20-0258-0	TNL to S/360 Continuous System Modeling Program (360A-CX-16X) System Manual Re: Y20-0111-0	NEW
Y20-0259-0	DP Sales Manual Labels	NEW
Y28-2014-1	IBM S/360 Time Sharing System Program Control System PLM	Scrap Y28-2014-0
Y28-2015-3	IBM S/360 Time Sharing System System Generation & Maintenance PLM	Scrap Y28-2015-2
Y28-2022-0	IBM S/360 Time Sharing System Time Sharing Support System PLM	NEW
Y28-3084	IBM S/360 Time Sharing System Access Methods (PLM) Re: Y28-2016	NEW
Y28-3087	IBM S/360 Time Sharing System FORTRAN IV PLM Re: Y28-2019	NEW
Z19-0001-0	IBM S/360 Direct Access Storage Devices Marketing Guide Section 1	NEW - Scrap Z20-0002
Z20-0831-1	Systems Reliability Prediction Technique - SRPT (360A-SE-34R) Program Reference Man.	NEW
Z20-1891-0	IBM S/360 Model 20, Application in Foundries Sales & Systems Guide	NEW
Z77-8082-0	Molins System 24 - The second Industrial Revolution	NEW
Z77-8088-0	DOS Version III System Generation Using the Private Library Facility	NEW
Z77-8090-0	Development of an Operators Training Class for Operating S/360	NEW
Z77-8091-0	Operating System/360 Device Independence Considerations	NEW

October 18, 1968 Issue No. 68-20

IBM INSTALLATION NEWSLETTER

TABLE OF CONTENTS

OS/360 DATA SET NAMING

Data set names that begin with the letters SYS and have a P as the nineteenth character of the name should not be used. Data sets with such names are deleted when the IEHPROGM utility is used and the SCRATCH utility control statement is specified with the VTOC and SYS keywords.

OS/360 MESSAGES TO THE OPERATOR

The following contribution has not been submitted to any formal IBM test. Potential users should evaluate its usefulness in their own environment, including maintenance, prior to implementation.

Under OS/360 MVT, messages to the operator are buffered using main storage in the system queue area (SQA) for buffers. Both the maximum number of buffers (at 144 bytes each) and the initial size of the SQA are specified when the system is generated, using respectively the WTOBFRS parameter of the SCHEDULR macro instruction and the QSPACE parameter of the CTRLPROG macro instruction. Default values are 100 buffers and a 20K SQA initially. (The size of the SQA may be set to a larger value when the Operating System is loaded; thereafter, if additional space is required, the SQA will expand in 2K increments.)

The program shown in Exhibit 1 provides an example of a technique for monitoring the number of buffers actually used at an installation. Several factors influence the maximum number of message buffers required at any one time. One, of course, is the number and frequency of WTO macro instructions issued by the programs being run. Another factor is the number of volume mounting messages generated by the system to fulfill the device allocation requirements of the JCL it is processing. A third is the number of jobs in the system when the operator issues a DISPLAY N command (including jobs waiting to be initiated and those which have been executed but whose output has not yet been processed by an output writer, as well as those currently running). A fourth is the number of tasks active when a DISPLAY A command is issued.

Frequent use of these DISPLAY commands is recommended in the SRLs to provide the operator with status information. Note, though, that in addition to generating a string of messages which occupy the buffers, that when the system is responding to a DISPLAY command and is also trying to issue an operator-intervention message (such as a volume mounting instruction), the action message will not be written to the console until all lines of output have been written in response to the DISPLAY command. Having only a few jobs in the system or a few active tasks can cause a delay of twenty seconds or more until the action message is seen by the operator.

The program is useful for monitoring the number of buffers required at a particular installation, taking into consideration all of the above factors. Running it in conjunction with both a user's workload and his operating procedures (such as the frequency of issuing DISPLAY commands and the number of reader, initiator, and writer tasks he starts) will provide data that could be used to calculate the parameter values to specify at system generation time and to devise better operating procedures. Sample JCL to run the program is shown within the program listing.

This program has been used to demonstrate (a) that the use of the various DISPLAY commands drastically increases the extent of the backlog of messages to the console, (b) that if sufficient buffers are not provided to handle the backlog, the time to process a jobstream is increased proportionately to the amount of typing wait time introduced, (c) that a guideline for an adequate number of buffers is: three times the number of active initiators plus the maximum number of jobs expected to be in the system at any one time, and (d) that the consequence of allowing too many buffers is that the SQA will attempt to expand to accommodate them as they are filled, but if the SQA cannot expand (the next adjacent 2K area of storage is in use by a task in the dynamic region) the system is put into a wait state and must be reloaded.

OS/360 MESSAGES TO THE OPERATOR

```
WTOCHEK  CSECT
*        WRITE TO OPERATOR BUFFER UTILIZATION MONITOR - WTOCHEK
*
*        THIS PROGRAM IS DESIGNED TO RUN UNDER OS/360 MVT. IT
*        RUNS AS A BACKGROUND JOB DURING THE EXECUTION OF A
*        CONTROLLED JOBSTREAM FOR THE PURPOSE OF ACCESSING THE
*        CELLS CONTAINING THE CURRENT WTO MESSAGE AND REPLY BUFFER
*        COUNT. ALL STATISTICAL INFORMATION IS PRINTED OUT WHEN THE
*        JOBSTREAM BEING RUN HAS COMPLETED AND ALL INITIATORS
*        HAVE BEEN STOPPED.
*
*        INPUT/CONTROL STATEMENTS
*              PARM FIELD OF EXEC STATEMENT MUST CONTAIN A 4 DIGIT
*              DECIMAL NUMBER REPRESENTING THE TIME INTERVAL,
*              IN HUNDRETHS OF SECONDS, AT WHICH  @WTOCHEK@ WILL
*              RECEIVE CONTROL THROUGH AN INTERVAL TIMER INTERRUPT.
*
*              A DD STATEMENT SPECIFYING THE SYSOUT DATA SET FOR
*              OUTPUTTING STATISTICAL DATA$ DDNAME#WTODAT
*
*                   //STP1  EXEC PGM#WTOCHEK,PARM#@0200@
*                   //WTODAT  DD SYSOUT#A,DCB#¤LRECL#121,BLKSIZE#121*
*
*        OPERATOR MESSAGES-
*              A MESSAGE IS WRITTEN TO THE OPERATOR AT INITIALIZATION
*              AND IN THE EVENT THAT THE DATA STORAGE AREA OVERFLOWS.
*        PROCESS LOOP
*
*                   .@STIMER@- SET TIME INTERVAL TO RECEIVE CONTROL
*                              AT SET INTERVALS.
*                   . ACCESS WTO BUFFER COUNTS THROUGH CVT
*
*                   .       CVT.100 POINTS TO UNIT CONTROL MODULE ¤UCM*
*                            * * * * * * UCM  OFFSETS * * * * * * *  *
*                            *
*                            * OFFSET¤DEC*  SIZE     CONTENTS ¤BIN* *
*                            *           *           *             *
*                            *   UCM.52  * 2BYTES *CURRENT REPLY   *
*                            *           *         *BUFFERS IN USE  *
*                            *   UCM.54  * 2BYTES *CURRENT MESSAGE *
*                            *           *         *BUFFERS IN USE  *
*                            * * * * * * * * * * * * * * * * * * * *
*
*                   *  CVT.148 POINTS TO MASTER SCHEDULER RESIDENT
*                      DATA AREA
*
*                            MSRDA.42 CONTAINS 1 BYTE CELL ¤BASTPNIT*
*                            CONTAINING THE STOP INITIATOR COUNT
*
*                   . WHEN ABOVE CELL CONTAINS A COUNT GREATER THAN 0
*                     OR WHEN BUFFERS OVERFLOW THE GETMAIN BUFFER AREA,
*                     WTOCHEK COMPILES AND OUTPUTS ALL STATISTICS
*                   IT IS NECESSARY TO MANUALLY SET THE CELL TO ¤1*
*                   IN RELEASE 15 MVT OR LATER. IN PRIOR RELEASES, THE
*                   INITIATORS BEING STOPPED WILL SET IT TO 1 OR MORE
```

EXHIBIT I

```
*                    • STORE BUFFER COUNT DATA IN GETMAIN AREA
*                    • IF THE DATA COLLECTION AREA BECOMES FULL
*                      DURING EXECUTION, THE PROGRAM WILL AUTOMATICALLY
*                      PURGE ALL DATA AND TERMINATE. THE AREA USED FOR
*                      DATA COLLECTION SHOULD BE INCREASED IN SIZE
*               OUTPUT
*
*                    • REPORT OF BUFFER UTILIZATION
*
*                        * INTERVAL*   WTO BUFFERS IN USE *
*                          NUMBER      MESSAGE     REPLY
*                        *           *           *          *
*                            1           01          01
*                        *   2       *   03      *   00     *
*                            3           06          02
*                        *   4       *   04      *   02     *
*                            •           •           •
*                        *           *           *          *
*                            •           •           •
*                        *           *           *          *
*                    • CLOSE SYSOUT
*
*                    • REINITIALIZE AND CONTINUE OR TERMINATE UPON
*                      OPERATORS REPLY.
*
*               REQUIRED CONFIGURATION AND REGION
*
*                    • INTERVAL TIMER
*                    • 30K MINIMUM REGION ¤25KGETMAIN ALLOWS
*                                         WTOCHEK TO GAIN CONTROL
*                                         12,500 TIMES TO MONITOR
*                                       BUFFER COUNTS*
*
*
*
R0        EQU   0
R1        EQU   1
R2        EQU   2
R3        EQU   3
R4        EQU   4
R5        EQU   5
R6        EQU   6
R7        EQU   7
R8        EQU   8
RB        EQU   12
RSAV      EQU   13
*
*
          SAVE  ¤14,12*,,WTOCHEK
          BALR  RB,R0
          USING *,RB
          ST    13,SAVAR.4
          LR    10,13
          LA    13,SAVAR
          ST    13,8¤10*
```

Exhibit I Continued

```
            L     R2,0¤R1*             GET TIME INTERVAL FROM
            PACK  WKAR1,4¤4*,2¤4,R2*   PARM FIELD AND
            CVB   R3,WKAR1             CONVERT FOR TIME
            ST    R3,TIMINT                 INTERVAL MACRO
            L     R0,MAIN
            GETMAIN  R,LV#¤0*          GET STORAGE
            ST    R1,DATBUF               FOR DATA
            ST    R1,BUFPTR            INITIALIZE START PTR
            L     R2,MAIN                    COLLECTION
            AR    R2,R1
            ST    R2,ENDBUF
*           INITIALIZE POINTER TO CELLS IN UCM
            L     R2,X@10@             CVT POINTER
            L     R3,100¤R2*           UCM POINTER
            LA    R4,52¤R3*            BUFFER COUNT CELLS IN UCM
            ST    R4,UCMPTR            SAVE POINTER
            L     R3,148¤R2*           MS RESIDENT DATA AREA
            LA    R4,42¤R3*            ADDR OF 1BYTE STOP INITIATOR COUNT
            ST    R4,SITCTR            SAVE POINTER
*
*           WTO  MESSAGE   AND ENTRY FROM SYSOUT PROCESSING
*
WTOC1       EQU   *
*           NOTIFY OPERATOR OF STATUS AND WAIT FOR REPLY
            WTO   @BEGIN WTO BUFFER ANALYSIS AT SET INTERVAL@
*
*           BEGIN ANALYSIS LOOP        YES
*
WTOC2       LM    R1,R4,BUFPTR
*           RETRIEVE INFORMATION FROM UCM
            MVC   0¤1,R1*,1¤R3*        MOVE REPLY BUFFER COUNT
            MVC   1¤1,R1*,3¤R3*        MOVE MESSAGE BUFFER COUNT
            LA    R1,2¤R1*
            ST    R1,BUFPTR            SAVE NEW BUFFER POINTER
            CLR   R1,R2                END OF AVAILABLE SPACE
            BNL   WTOC3                YES
            CLI   0¤R4*,X@00@          NO. INITIATORS STOPPED. END OF J/S.
            BNE   WTOC31               YES. PREPARE FOR PRINTOUT
            STIMER WAIT,BINTVL#TIMINT        NO. SET TIMER AND WAIT
            B     WTOC2
*
*           OUTPUT ROUTINE- ALL DATA IS OUTPUTTED AT THE END OF A RUN OR
*           WHENEVER THE BUFFERS ARE FULL.
*
WTOC3       LR    R6,R1
*           NOTIFY OPERATOR OF END OF AVAILABLE BUFFER SPACE
            WTO   @DATA COLLECTION AREA FULL, PURGING TO SYSOUT @
*
*           OUTPUT DATA TO SYSOUT
*
            B     WTOC311
WTOC31      LR    R6,R1                LAST DATA STORED
WTOC311     EQU   *
            OPEN  ¤DCBOUT,¤OUTPUT**    READY OUTPUT D/S
            LA    R8,1
```

Exhibit I Continued

```
            L       R5,DATBUF             TOP OF DATA AREA
WTOC32      XC      WKBUF,WKBUF
            XR      R3,R3
            NI      LSW,X@FE@
            MVC     WKBUF□50*,HDR         SET UP HEADER
            PUT     DCBOUT,WKBUF
            L       R7,LINCNT             LINE COUNTER
WTOC33      XC      WKBUF,WKBUF
            IC      R3,0□R5*              PICK UP REPLY COUNT
            CVD     R3,WKAR1              CONVERT COUNT
            UNPK    WKBUF.44□3*,WKAR1.6□2*    FOR
            OI      WKBUF.46,X@F0@              OUTPUTTING
*
            IC      R3,1□R5*              PICK UP MESSAGE BUFFER COUNT
            CVD     R3,WKAR1              CONVERT COUNT
            UNPK    WKBUF.26□3*,WKAR1.6□2*   FOR
            OI      WKBUF.28,X@F0@             OUTPUTTING
            CVD     R8,WKAR1              CONVERT INTERVAL COUNTER
            UNPK    WKBUF.4□5*,WKAR1.4□4*
            OI      WKBUF.8,X@F0@
            LA      R8,1□R8*              UPDATE INTERVAL COUNT
            BCT     R7,NEWLIN
            OI      LSW,X@01@             SET FOR NEW PAGE
*
NEWLIN      PUT     DCBOUT,WKBUF          OUTPUT RECORD
            LA      R5,2□R5*              UPDATE BUFFER POSITION
            CLR     R5,R6                 LAST SET OF DATA OUTPUTTED
            BNL     WTOTERM               YES
            TM      LSW,X@01@             NO. CONTINUE.. NEW PAGE
            BO      WTOC32
            B       WTOC33                NO. NEW LINE
*
*           TERMINATION ROUTINE
*
WTOTERM     L       R0,MAIN               SIZE
            L       R1,DATBUF             ADDRESS
            FREEMAIN R,LV#□0*,A#□1*
            CLOSE DCBOUT
            L       13,4□13*
            SR      15,15
            ST      15,16□13*
            RETURN □14,12*,RC#□15*
*
*
WKAR1       DC      D@0@
WKAR2       DC      D@0@
MAIN        DC      F@12288@
SAVAR       DS      19F
DATBUF      DS      F
TIMINT      DS      F
BUFPTR      DS      4F
ENDBUF      EQU     BUFPTR.4
UCMPTR      EQU     BUFPTR.8
SITCTR      EQU     BUFPTR.12
LINCNT      DC      F@48@
```

Exhibit I Continued

OS/360 MESSAGES TO THE OPERATOR

```
WECB      DC    F@0@
WREP DC F@0@
LSW       DC    X@00@
HDR       DC    C@1INTERVAL NO.      MESSAGE-BUFFERS     REPLY BUFFERS@
          DS    0D
WKBUF     DS    CL121
*
          DS    0F
DCBOUT    EQU   *
          DCB   DSORG#PS,MACRF#PM,EROPT#ACC,                          *
                RECFM#FSA,                                            X
                DDNAME#WTODAT
          END
/*
```

Exhibit I Continued

OS/360 SPECIFICATION OF FIFO OR PRIORITY QUEUEING FOR I/O REQUESTS

The IO REQUE option of the IO DEVICE SYSGEN macro specifies the type of I/O request queueing to be provided by the supervisor for the device. FIFO specifies first-in-first-out queueing. PRIORITY specifies queueing according to task priority. On a given channel, I/O requests for those devices which have been specified as PRIORITY will, as a group, have channel priority over I/O requests for those devices which have been specified as FIFO. Within the group of devices specified as PRIORITY, I/O requests are assigned channel priority according to the priority of the associated tasks. Within the group of devices specified as FIFO, I/O requests are assigned channel priority on a first-in-first-out basis.

PRIORITY queueing is most useful for devices associated with tasks which require high response and fast turn-around time. Teleprocessing applications are a group that might fall into this category.

It should be noted that an I/O bound task using devices which have been assigned PRIORITY queueing can monopolize the channel for an extended period. If this happens, lower priority tasks using PRIORITY devices on that channel, and all tasks using FIFO devices on that channel will have to wait and their throughput will be degraded.

It is recommended that a detailed study of its impact be made to justify the use of PRIORITY queueing in the normal batch environment.

OS/360 MSGLEVEL = 1 DEFAULT OPTION FOR MVT SYSTEMS CLARIFICATION

Installation Newsletter issue 68-17 contained, on page 2, an article on the above subject. The article contained some typographical errors and is reprinted below and replaces the article in issue 68-17.

The following contribution has not been submitted to any formal IBM test. Potential users should evaluate its usefulness in their own environment prior to implementation.

The Super-Zap fix shown below will allow a MSGLEVEL = 1, a default option. This avoids the need to punch MSGLEVEL =1 in a job card.

SYSLIB	DD	DSNAME=SYS1.LINKLIB
NAME	IEFVHA	IEFVJA
VERIFY	0036	D100, 8007, C019
REP	0036	921x, 8007, 0700

Where "x" is the hexadecimal default priority specified in the reader procedure for jobs. The above has not been checked against the coding for MFT-II.

OS/360 RELEASE 16 FORTRAN GSP SPEC A GRAPHIC SUBROUTINE ADDITION

With OS/360 Release 16 the FORTRAN user has the ability to change the pre-defined LINK-LOAD status of GSP (Graphic Subroutine Package 360S-LM--537) subroutines. Use of this subroutine can optimize the core vs. speed tradeoff.

For example, the GSP subroutine PSGMT generates the graphic orders to draw vectors on the 2250 screen; the pre-defined status for this routine is LOAD. If the programmer CALL's PSGMT only once in his program it would enhance overall performance to LINK to the routine; the copy would then be deleted after the vector orders were generated, resulting in additional core being available. SPEC may be CALL'ed at any point in the program and as many times as desired. Therefore, through the use of SPEC, a significant improvement in performance may be realized. See SRL OS/360 GPS for FORTRAN IV (C27-6932) for more information.

OS/360 PL/I ISAM FILE PROGRAMMING TIP

Even though ISAM files always have recorded keys, if the KEY or KEY TO option is not used on a READ, DO NOT specify the KEYED attribute for SEQ INPUT and/or UPDATE files. This will reduce core requirements by keeping the QISAM SETL routine from being loaded at OPEN time. The storage saved can be approximated by the following formula:

$$1816 + 320_n$$

where: n = number of indexed files opened.

OS/360 CREATING MULTIPLE PDS FILES ON A VOLUME WITH IEHMOVE

Multiple PDS files on a single volume - The Utility Program IEHMOVE will create a second or subsequent data sets on a standard label tape if the correct utility control statements are used. The data set sequence number on a tape volume must be specified in the utility control statement as specified in the Identification Parameters section of the Utilities SRL, pages 24-25 form C28-6586-8, and should not be specified in the JCL TO DD statement. However, this point is not clearly defined in the Utilities SRL and a mistake could be made.

OS/360 ADVANCED OPERATOR TRAINING COURSE

The S/360 OS Advanced Operator Training Self-Instruction--is now available. This course is provided for System Operations personnel who perform the specific function of analyzing operational situations requiring corrective action and executing such action. It is intended to be conducted on the user's site. The course consists of a series of PI and Student Texts which the student will read before performing hands-on exercises or participating in scheduled group discussions. The complete guideline for the conduct of this course and the recommended sequence of modules is contained in the Operator Training Discussion Guide (R20-4122) which also outlines the topics of the planned discussion sessions. This guide will be used by the course administrator who should be fully qualified in OS Operations and capable of monitoring this training program.

All elements of the course may be ordered by the IBM Branch Office from Mechanicsburg on standard requisition forms. The number of the course description is R20-9244-0, and the number of the Discussion Guide which details the course conduct is R20-4122.

List of Course Elements

A. For the Student:

System/360 OS Concepts for Operators

Learner's Guide Text	R20-4114
Learner's Guide Illustrations	R20-4115

JCL for Operators

Learner's Guide Text	R20-4118
Learner's Guide Illustrations	R20-4119

Utilities for Operators

Learner's Guide Text	R20-4120
Learner's Guide Illustrations	R20-4121

System/360 Assembler Language Coding

System Review Text (PI)	R29-0231

Depending upon student experience, the following may also be required:

Computing Systems Fundamentals PI course

Unit 1	R29-0241
Unit 1 Notebook	R29-0242
Unit 2	R29-0243
Unit 3	R29-0244
Unit 4	R29-0245
Unit 5	R29-0246
Unit 6	R29-0247
Units 2-6 Notebook	R29-0248

OS Systems Operation

Text	C20-1680
Illustrations	C20-1681
Exercise	V25-6463

B. For the Course Administrator (in addition to above)

System/360 OS Operator Training

Discussion Guide	R20-4122
Operator Training Mini-Reel	R20-4123

DOS/360 COBOL 1400 SERIES LABEL PROCESSING

The following is a field contribution which has not been submitted to any formal IBM test. Potential users should evaluate its usefulness in their own environment prior to implementation.

When creating 1460 tapes with a S/360 DOS COBOL program, the following procedures will eliminate the tapemark on the output tape and write an 80 character label at the beginning of the tape. Note that this procedure does not do any label checking nor does it detect wrong length records that are less than the maximum record length.

```
FD   TAPE-IN
     RECORD CONTAINS 8Ø TO 12Ø CHARACTERS
     RECORDING MODE IS U
     LABEL RECORDS ARE OMITTED
     DATA RECORDS ARE RECORD-A, RECORD-B.

01   RECORD-A      PICTURE  X(8Ø).
Ø1   RECORD-B      PICTURE  X(12Ø).

FD   TAPE-OUT
     RECORD CONTAINS 8Ø TO 12Ø CHARACTERS
     RECORDING MODE IS U
     LABEL RECORDS ARE OMITTED
     DATA RECORDS ARE RECORD-1, RECORD-2.

01   RECORD-1      PICTURE  X(8Ø).
Ø1   RECORD-2      PICTURE  X(12Ø).

PROCEDURE DIVISION.
     ENTER LINKAGE.
     CALL 'NOMARK' USING TAPE-OUT.
     ENTER COBOL.
```

```
        OPEN INPUT TAPE-IN, OUTPUT TAPE-OUT.

        READ TAPE-IN AT END GO TO END-ROUTINE.

        WRITE RECORD-1 FROM RECORD-A.
```

The NOMARK routine modifies Byte 32, bit 6 of the DTF for the tape output file, so that a tape mark is not written on the output file.

```
// JOB NOMARK
// OPTION DECK
// EXEC ASSEMBLY

NOMARK   START   Ø
         ENTRY   NOMARK
         SAVE    (14,12)
BEGIN    BALR    1Ø,Ø
         USING   *,1Ø
         L       2,Ø(Ø,1)       LOAD DTF ADDRESS
         OI      32(2),X'Ø2'    SET ON BIT 6
         RETURN  (14,12)
         END
/*
/&
```

S/360-20 RPG USE OF MR INDICATORS ON OUTPUT

The following is a field contribution. It illustrates the incorrect and correct use of MR Indicator on Output Specifications in conjunction with card type indicators. See Exhibit 1.

Example 1.

> 01 and 02 are secondary file card type indicators. With this coding, the first detail card of a detail group is not punched with information from a matching master card. All other detail cards that match the master are punched correctly.

Example 2.

> All matching detail cards are punched correctly.

S/360 RPG ROUTINE FOR AGING CORRECTION

Installation Newsletter issue 68-11, dated June 14, 1968 contained an article, on page 7, with the above title. The author has provided corrections and replacement Calculation Specifications as shown in Exhibit 1. This article replaces the previous one contained in issue 68-11.

The following field contribution has not been submitted to any IBM test. Potential users should evaluate its usefulness in their own environment prior to implemenation. Exhibit 1 is an example of an RPG routine for aging. Two input fields (which allow for a difference of up to 999) are used:

1. A control date (CURDAT) - YYMM
2. A transaction date (TRNDAT) - YYMM

Two changes have been made to the Calculation Specifications which added:

1. Indicators for the current month and not yet due have been added.

2. Factor 1 has been added to the compare instructions.

IBM

INTERNATIONAL BUSINESS MACHINES CORPORATION

REPORT PROGRAM GENERATOR OUTPUT-FORMAT SPECIFICATIONS

IBM System 360

Form X24-3352-1 U/M 025
Printed in U.S.A.

Date _____

Program _____

Programmer _____

Punching Instruction	Graphic							
	Punch							

Page 1 2 *

Program Identification 75 76 77 78 79 80

EXHIBIT I

Line 3 4 5	Form Type 6	Filename 7 8 9 10 11 12 13 14	Type (H/D/T) 15	Stacker Select 16	Space Before 17	Space After 18	Skip Before 19 20	Skip After 21 22	Output Indicators And/Not 23 24 25	And/Not 26 27 28	And/Not 29 30 31	Field Name 32 33 34 35 36 37	Zero Suppress (Z) 38	Blank After (B) 39	End Position in Output Record 40 41 42 43	Packed Field (P) 44	Constant or Edit Word 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70	Sterling Sign Position 71 72 73 74
0 1	O	OUTPUT	D						MR									
0 2	O								Ø1			FIELDA			6Ø		(1)	
0 3	O								Ø2			FIELDB			6Ø			
0 4	O																	
0 5	O	OUTPUT	D						MR	Ø1								
0 6	O		OR						MR	Ø2							(2)	
0 7	O								Ø1			FIELDA			6Ø			
0 8	O								Ø2			FIELDB			6Ø			
0 9	O																	
1 0	O																	
1 1	O																	
1 2	O																	
1 3	O																	
1 4	O																	
1 5	O																	
	O																	
	O																	
	O																	
	O																	
	O																	

October 18, 1968

Issue No. 68-20

S/360 RPG ROUTINE FOR AGING CORRECTION

IBM

INTERNATIONAL BUSINESS MACHINES CORPORATION

REPORT PROGRAM GENERATOR CALCULATION SPECIFICATIONS

IBM System/360

Form X24-3351-1 U/M025
Printed in U.S.A.

Date _____

Program _____

Programmer _____

Punching Instruction	Graphic							
	Punch							

Page 1 2

Program Identification 75 76 77 78 79 80

Line 3 4 5	Form Type 6	Control Level (L0-L9, LR) 7 8	Indicators: And / And — Not 9 10 11	Not 12 13 14	Not 15 16 17	Factor 1 18 19 20 21 22 23 24 25 26 27	Operation 28 29 30 31 32	Factor 2 33 34 35 36 37 38 39 40 41 42	Result Field 43 44 45 46 47 48	Field Length 49 50 51	Decimal Positions 52	Half Adjust (H) 53	Resulting Indicators: Plus 54 55	Minus 56 57	Zero or Blank 58 59	Comments 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74
													Compare: High 1 > 2	Low 1 < 2	Equal 1 = 2	
01	C					CURDAT	SUB	TRNDAT	DIFF	3Ø						
02	C					DIFF	COMP	88					5Ø			
03	C		5Ø			DIFF	SUB	88	DIFF							
04	C					DIFF	COMP	Ø						48	49	48=NOT YET DUE
05	C		48				GOTO	NEXT								49=CURRENT
06	C		49				GOTO	NEXT								
07	C					DIFF	COMP	1							51	3Ø DAYS OVER
08	C		51				GOTO	NEXT								
09	C					DIFF	COMP	2							52	6Ø DAYS OVER
10	C		52				GOTO	NEXT								
11	C					DIFF	COMP	3							53	9Ø DAYS OVER
12	C		53				GOTO	NEXT								
13	C					DIFF	COMP	4							54	12Ø DAYS OVER
14	C		54				GOTO	NEXT								
15	C					DIFF	COMP	5					56		55	55=15Ø DAYS OV
16	C					NEXT	TAG									56=18Ø DAYS OR MORE
	C															
	C															
	C					CONTINUE with detail calculations										
	C															

EXHIBIT I

Card Electro Numbe

October 18, 1968

Issue No. 68-20

NEW TYPE III PROGRAMS

The following are the abstracts of programs which have been recently made available from the Type III library.

The program, along with its complete abstract, will be incorporated in subsequent issues of the Catalog of Programs.

Programs may be obtained by submitting a properly completed "General Program Request Card" (Form Number 120-1145) to the Program Information Department, 40 Saw Mill River Road, Hawthorne, New York, 10532.

These programs and their related documentation are distributed by IBM in the author's original form and have not been subjected to any formal testing.

Any discussions of Type III programs with customers must emphasize the following points:

1. Type III (IBM employee-contributed) programs are provided by the IBM Corporation as part of its service to customers.

2. Type III programs have not been subjected to any formal test.

3. Recipients of Type III programs are expected to make the final evaluation as to the usefulness of the programs in their own environment.

4. There is no committed maintenance for Type III programs. However, any changes the author chooses to make will be announced in subsequent issues of the Catalogs of Programs.

5. IBM makes no warranty, expressed or implied, as to the documentation, function, or performance of Type III programs.

*NOTE: THE CUSTOMER MUST BE INFORMED THAT THE ABOVE APPLIES TO ANY OF THE FOLLOWING TYPE III ABSTRACTS FURNISHED TO HIM.

---REVISION---

MODEL 20 RPG/BAL RELOCATABLE FEDERAL WITHHOLDING TAX ROUTINES. Revised documentation and machine readable material.

Ordering Procedure: Order File Number 360E-21.1.001

The program material can be obtained in card form or on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

If not specified, a 9 track DTR will be provided. No tape submittal is required. The DTR will be provided by the Library.

---REVISION---

ARRAY PROCESSOR ACCESS METHOD FOR IBM SYSTEM 360 MODEL 44.

Revised program documentation and machine readable material.

Ordering Procedure: Order File Number 360D-03.4.019.

The program material can be obtained on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

If not specified, a 9 track DTR will be provided. No tape submittal is required. The DTR will be provided by the Library.

HOUSTON AUTMATIC SPOOLING PRIORITY SYSTEM-II. The HASP SYSTEM is an automati SYSIN/SYSOUT/SYSPUNCH SPOOLing program with the ability to operate an essentially unlimited number of peripheral devices (including high-speed remote batch terminals) concurrently in conjunction with OS/360 processing. A variable number of 2311 and/or 2314 direct access modules are utilized for intermediate storage. Because of the specialized, singular use of resources available to HASP, very efficient algorithms have been employed to optimize function and performance. Any standard OS/360 job can be processed under a system utilizing HASP with no changes to the deck, and can make full use of the programming capability and flexibility offered by OS/360. HASP will operate inter-changeably with either the MFT or MVT options of OS/360-Release 15/16, but has been designed to be as release independent as possible to permit adaptation to future Operating System releases. No modifications are required to OS/360. HASP can be utilized on any System/360 capable of supporting either of the aforementioned options of the Operating System. The program source is in OS/360 Assembly Language and has a minimum storage requirement of approximately 36K.

Ordering Procedure: Order File Number 360D-05.1.014.

To obtain program material submit one 2400 foot reel of magnetic tape. Specify whether 7 or 9 track recording is required. If not specified, 9 track recording will be used.

The required tape may be ordered from IBM or supplied with your request for the program.

MODIFIED COMPUTER PROGRAMS QUANTITATIVE ANALYSIS WITH ELECTRON MICROPROBE ANALYZER. A description of two FORTRAN IV computer programs is presented to simplify quantitative analysis with the Electron Microprobe Analyzer. The first program, EPMP1, determines the weight fraction of each element in a specimen from the characteristic X-ray intensity measurements of the specimen and the standards. The second program, EPMP2, calculates the relative characteristic X-ray intensities of all the elements in a specimen by assuming the composition of the specimen to be known. The correction procedure includes dead time correction, one of two background corrections (either constant background or background depending on composition), Philibert's absorption correction modified by Duncumb and Shields, one of three fluorescence corrections (either Birks', Castaing's, or Reed's), and a compound standard correction. The effects of the absorption and the fluorescence of each element in the specimen are easily seen from the output results. Versatility, efficiency, and ease of operation are emphasized in the programs. Program listings, input data format, and various examples showing the usage of the programs have been included in the Appendices. The programs operate under IBM S/360 Operating System with FORTRAN (level H) compiler and require a minimum core storage of 36K Bytes.

Ordering Procedure: Order File Number 360D-17.1.001.

The program material can be obtained in card form or on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

If not specified, a 9 track DTR will be provided. No tape submittal is required. The DTR will be provided by the Library.

KWIC/360- A KWIC AND KWOC INDEXING PACKAGE FOR SYSTEM/360. A package of four programs has been prepared which will accept data punched in the standard KWIC format as described in Form No. E20-8091 and will produce, at the user's option (a) Bibliography, (b) Author Index, (c) Permuted Title Index - KWIC format, (d) Permuted Title Index - KWOC format, (e) Assigned Descriptor Index. Most of the features of the 1401 program will be provided with, in addition, the ability to delete and replace entries in the bibliography, together with a more comprehensive list of stopwords, up to 255 to be nominated by the user. The programs are written in PL/1 with Assembler Language subroutines to run under OS/360.

KWIC/360 runs as one or more jobs under control of Operating System/360. It is necessary to include the PL/I Compiler when the system is generated. This is to ensure that the PL/I error handling package, which is called dynamically when it is needed, is present in the system Link Library. The largest program is KWICA with a Load Module of about 32,000 bytes, plus automatic variables occupying a further 8,000 bytes. Precise storage will

depend on options selected, blocking factors, and number of buffers allocated to the various files. It should be possible to run this program in a 44K partition, but many applications will require more. Other programs should run in 44K unless very large blocking factors are used.

All programs in the KWIC/360 package are re-entrant.

Ordering Procedure: Order File Number 360D-06.7.014.

The program material can be obtained on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

If not specified, a 9 track DTR will be provided. No tape submittal is required. The DTR will be provided by the Library.

PRIORITY OUTPUT WRITERS EXECUTION PROCESSORS, AND INPUT READERS. This DOS system is an automatic SYSIN/ SYSLST/SYSPCH spooling processor which requires no IBM or user program changes. Using a variable number of 2311 or 2314 modules for intermediate storage, this program acts as a peripheral processor and priority scheduler for normal batch processing under DOS. It can, simultaneously, operate up to 26 pheripheral devices, maintain unique input/ print/punch or print/punch streams for one or two DOS batch partitions.

Machine Configuration: any hardware which support DOS; practically, however, 32K or 48K would be minimum core.

Programming SYSTEMS: Standard DOS.
Source Language: ASSEMBLY LANGUAGE
Storage Requirements: varies based on SYSGEN options; minimum would be 12K.

Ordering Procedure: Order File Number 360D-05.2.006.

The program material can be obtained on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

If not specified, a 9 track DTR will be provided. No tape submittal is required. The DTR will be provided by the Library.

COLORANT DATA PLOTTER. The purpose of this program is verification of primary colorant data used in color matching programs such as the Type III 1130-16.3.001. The program produces a graph of K/S (K/S is a mathematical expression of the characteristics of a color and is a function of the percent of reflected light incident upon a colored sample) vs. wavelengths of light within the visible spectrum. These curves of K/S vs. wavelength are produced for up to nine concentration levels of a primary colorant and can be analyzed for accuracy.

Also produced is a plot of K/S vs. concentration at the point of maximum absorption, which should approach a linear relationship, and which can be analyzed for linearity as well as for accuracy.

The program is to be supplied in FORTRAN source form. The required configuration is at least an 8K 1130, DISK, 1442, and either TYPEWRITER or 1132 PRINTER output.

Ordering Procedure: Order File Number 1130-16.3.002.

Distribution will be in card form only.

A GENERALIZED INDEXED, SEQUENTIAL MAIN FILE WITH LINKED OVERFLOW APPLICATION FOR 1130/1800 USERS. This set of programs is meant to give the 1130/1800 user an index sequential type file organization that is as general in its applications and as easy to use as possible. It consists of five programs to create, process randomly, process sequentially, add to and delete from the file. An attempt has been made to use and expand upon the ideas and methods in the disk file organization and processing portion of the 1130 Commercial Programming Education Guide.

Each of the programs is a main-line program, but can be modified to be a subroutine. There are a minimum number of changes to be made to modify them to apply to different files or systems. All are written in FORTRAN to simplify user modification.

Ordering Procedure: Order File Number 1130-06.7.002.

Distribution will be in card form only.

IBM Installation Newsletter

December 13, 1968 Issue No. 68-24

CONTENTS

(C means Circular Section)

Distribution: Branch Office - DP Management, Salesmen, Systems Engineers, FE Managers, Regions, Districts, Education Centers, Field Systems Centers, Federal Systems Centers, SE Area Office, DPD HQ, FED HQ, WTC.

*Requires Immediate Attention.

DPD HQ, White Plains, N. Y.

SELECTED SHORT SUBJECTS

The purpose of Selected Short Subjects is to bring together and highlight concise, factual and timely information which will indicate that action is to be taken by the IBM representative whose accounts are affected.

1. QUESTER-Reducing Duplication of Effort

Are you about to develop a special program modification or routine? Would you like to find someone who is considering or has already satisfied a similar requirement? If so, have you considered QUESTER?

QUESTER is designed to reduce duplication of effort. It may help you!

QUESTER (QUEstion or Solution To Eliminate Redundancy) is an internal IBM system which provides for the exchange of information between Branch SE Managers. This information source, originally developed by the regions, is now coordinated by DP Systems Engineering, DPD HQ.

QUESTER data consists of statements of special local requirements and branch development activities which may be of general use but are not planned to result in IBM Type III or Type IV programs or TIE papers.*

Monthly, a KWIC Index with abstracts of DP and WTC branch office efforts and needs is prepared and distributed to each SD&I Center.

To take advantage of this information exchange, your SE Manager can submit a QUESTER form stating your need or planned project to your SD&I Center. He in turn will be advised of other locations having similar needs or completed activities. As your project is completed, it should also be reported using the QUESTER forms so that others may benefit from your efforts. If your branch office is out of forms, additional padded sets may be obtained through your SD&I Center or Regional QUESTER co-ordinator.

*SECOM, the Catalogs of Programs, and the KWIC Index to TIE identify these activities.

OS/360 TRACE TABLE CONSIDERATIONS

Inclusion of the optional trace table is recommended in all systems where available core storage allows this to be done. The trace table contains valuable debugging information pertaining to SIO instruction executions, SVC interruptions and I/O interruptions.

The user specifies the number of entries desired by the TRACE option of the SUPRVSOR SYSGEN macro. The entries are 16 bytes long for PCP and MFT II, and 32 bytes long for MVT.

For the trace table to be generally useful, the recommended minimum is 250 entries.

If a user desires the advantages of a trace table, but cannot spare the core storage, he can use alternate approaches. One alternate approach is to generate his system with both a primary and an alternate nucleus. The primary nucleus would be used for normal operation, and would not contain a trace table. The alternate nucleus would contain a trace table of sufficient size and would be used only when the trace table facility is needed.

A second alternate approach would be to generate a trace table with the minimum number of entries and then expand the table when it is needed. Field Engineering has a new program which can be used to expand the trace table after the system has been IPL'ed. Check with local Field Engineering as to the possibility of their making this Service Aid program available for this purpose. The program name is EXTEND and it is available as PTF 01100-00.

DOS/360 COMPATIBILITY SUPPORT/40, TYPE I

The following information is provided for planning purposes only. The actual availability and specification will be as provided through the formal IBM procedures.

The purpose of this article is to stress the importance of detailed planning, which is a must before installing Compatibility Support/40 (CS/40), Type I, announced in letter P68-99. After availability of CS/40, scheduled for

release April 30, 1969, PID will not distribute the Type III COS programs. There will also be no maintenance or modifications for installed Type III COS accounts. The following are requirements for the operation of CS/40 on a Model 40 other than those required for DOS:

1. 1401/1460 Compatibility (feature #4457).
2. 1401/1440/1460 DOS Compatibility (feature #4460).
3. 1311 Disk Compatibility (feature #9710).

Notice that feature #9710 must be installed regardless of the emulated 1400 configuration. Also, most important to notice is that feature #4460 precludes the use of the following:

1. Program 360C-EU-074 (Model 40 emulator).
2. Program 360D-05.1.101 (COS-40 Type III).
3. 1401/1460 stand-alone ability of feature #4457.

Conversion from emulation or COS-40 Type III to CS/40 Type I should not be attempted without extensive testing. There is now an attempt to have the stand-alone emulator modified so that it can coexist with CS/40. Until this is done, testing will have to be accomplished at other than the customer's location since installation of feature #4460 precludes the emulation of 1400 programs in the environment in which they are now being run. One of the key ground rules for Type I development was that external transparency be maintained. Therefore, CS/40, Type I, should not cause undue conversion problems from COS-40, Type III, if the Type III program has not been modified to meet specific customer needs.

There are no plans for customer education on either CS/30 or CS/40. It is expected that a lecture will be added to CST to provide some formal training for IBM people. With the existing field body of knowledge, this should be sufficient to meet our needs.

DOS/360 BTAM PRINTING ON 1053 ATTACHED TO LOCAL 2848/2260 CORRECTION

Articles on this subject were published in Installation Newsletter issues 68-12 and 68-15. It has been suggested that the Timings were incomplete and that the following statement should be used:

Timing estimates for 1053 print operation can be determined as follows:

$$\frac{\text{*Character count} + \text{length of carrier return (inches)}}{15} =$$

Time in seconds per line.

*Character count should include shift changes (if applicable), spaces and carrier return.

For the second article, the word "message" is suggested as a better choice than the word "line", since a write can be many lines.

S/360 MODEL 20-5 CORE REQUIREMENTS

Care must be taken when planning for the installation of the S/360 Model 20 submodel 5 CPU for existing DPS and TPS customers. Customers and salesmen should not plan on simply exchanging CPU's. There will be a difference in core usage both by the DPS and TPS monitor and by the program technique required to take advantage of the read/write/compute (R/W/C) capabilities. With the delivery of the 2152 Printer/Keyboard, the new DPS generative montior will be available. With the delivery of the 3620 submodel 5, the Read/Write/Compute (R/W/C) overlap capabilities will be available for both tape and disk systems. Under TPS, to take full advantage of R/W/C, the monitor will require approximately 2400 bytes. Add to this, additional I/O areas which utilize the double-buffering technique on a "per-file" basis to arrive at core usage estimates. Thus programs near capacity of existing systems will now need to be re-evaluated to determine the necessity of larger memory.

Under DPS the size of the monitor will vary, since the user will "generate" it based upon his needs. The following chart shows some examples of this:

Current size of monitor, without tape = 2858 bytes
Current size of monitor, with tape = 3518 bytes
New monitor, with 2 disks, no tapes and R/W/C and interrupt capabilities = 4532 bytes
New monitor, with 4 disks, 6 tapes, and R/W/C and interrupt capabilities = 4662 bytes
New monitor, with 1 disk, no tapes, and no R/W/C or interrupt capabilities = 2652 bytes

These estimates are taken from the tables in IBM Programming Announcement letter P68-85, dated June 17, 1968. Please refer to this letter for your individual estimates.

S/360 TIMER ACCURACY

The following is an expansion on an article on Timer Accuracy in Issue No. 68-20 of this Newsletter.

The interval of time which constitutes the precision of the timer on a System/360 is 1/60 of a second for a machine powered by 60 cycle A. C. and 1/50 of a second for a machine powered by 50 cycle A. C.

This imposes some limitations on the resolution of the timer.

1. A time interval smaller than 16 2/3 millisecond will produce a value of either 0 ms or 16 2/3 ms.

2. A time interval larger than 16 2/3 ms will produce a time that is an integer multiple of 16 2/3 ms.

The resolution of the timer can be improved by statistical means if the event can be timed several times.

An average can then be computed. However, the smaller the interval of time to be measured the more times the event should be timed in order to arrive at a suitable average.

The above times are true for a machine which is not equipped with a device which changes the rate at which the timer is updated. However, similar logic applies to that situation.

BTAM SYSTEM EXCHANGE OF INFORMATION BETWEEN PARTITIONS

Some BTAM users would like to design a BTAM system which operates in a two partition environment similar to QTAM. Control of the communication lines will be done by one partition and message processing by a second partition. The major advantage of this approach is that message processing is made subordinate to line control by placing the message processing program in a lower priority partition. This prevents message processing from having a detrimental effect on line control. The use of this technique implies the need for queuing of incoming and outgoing messages. More importantly, it implies the necessity for communication between the two partitions. BTAM, unlike QTAM, has no programmed facility for interpartition communication. There are, however, several techniques that the BTAM programmer can use for communicating between his two partitions the location of the next message on the queue or an indication that no messages are currently queued. The same techniques could be used for passing responses from the message processing program back to the line control program.

1. He may preformat his queue area on disk by placing some character (such as X'FF') which will never appear in message text into the first byte of every queue record. The message processing program may trace the entry of messages into the queue by periodically checking the first queue record to see if a message has overlaid the preset special character. Once the first queue record has been processed, it can continue processing until a queue record is found which still contains the preset special character. It must then go back to periodic checking of this queue record until it finds a message placed there.

2. Instead of formatting his queue records in advance, the BTAM programmer may decide to include in each message a field or fields which identify messages uniquely as part of today's queue. A combination of the current day's date and a message sequence number will normally suffice. The message processing program must again periodically check the next queue record to see if a message has been placed there.

3. The line control program may maintain a special record or records on disk which provide information on the current status of the queue or queues. Such a special record might contain such data as a pointer to the last message placed in the queue, the total number of messages placed on the queue, current status of lines and/or terminals, etc.

These first three techniques all require that the message processing program periodically look at disk records to see if new information has has been placed there. (This periodic checking

will probably be initiated by timer interrupts.) The major disadvantage here is that process time, channel time and disk unit time will be wasted each time that a nonproductive reference is made. Lengthening the time interval between references minimizes this overhead but has a detrimental effect on response time.

4. Information records such as those described in paragraph 3 may also be maintained by the line control program in core storage, probably at the upper end of core. Various message processing programs may access these records by referring to their actual core storage address. (Storage protection does not inhibit the message processing partition from reading data in the line control partition.) Using actual storage addresses in such cross-reference routines will generate the need for recoding if CPU storage size is changed. The impact of this problem can be minimized by writing a single library subroutine for such cross-references and link editing it into all message message processing programs.

5. The programmer may prepare user SVC routines to perform the required information exchange. One SVC would be required for use by the line control program. This SVC would contain space for information records and code for adding to or modifying these records. A second SVC would be required for the message processing program(s). It would provide code for reading the information records and passing results back to the inquiring message processing program. This technique is more easily implemented under OS than under DOS because OS provides the necessary "hooks" for the addition of user SVCs to the SVC library.

Both of the last two techniques still require a periodic checking by the message processing program to see if new entries have been placed in the queue. Thus, processing overhead is implied. However, since there is no requirement for I/O activity in connection with this testingof the queue, overhead is probably less than with the first three approaches. Also, response to the queue test is much more rapid.

None of these techniques has QTAM's QWAIT-QPOST capability of providing an interrupt to the message processing program on the arrival of the next message in the queue. Even though less sophisticated, they all do the required job. There should be at least one of these methods which is suitable for use with almost any type of BTAM based system.

1130 TYPE II PROGRAMS FOR OPERATING UNDER DISK MONITOR SYSTEM VERSION 2

All PLAN-based application programs use PLAN Direct Access Support, rather than FORTRAN Direct Access. They support additional 2310's without the use of the *FILE CARD option.

Users should be made aware that there is a severe restriction on core image using the *FILE CARD option. Variable location and variable file space allocation are not practical, using this combination. PLAN does support variable file allocation for core image.

TELEPROCESSING - DATA COMMUNICATIONS HANDBOOK IMPROVEMENTS

The following improvements to the Handbook have been made so far during 1968. All of these have been in response to field inquiries and suggestions. They can help you too.

Data Set Option Chart revised and expanded (Sec. 20/Area 16, Pages dated May 1968)

AT&T Tariff Abstract 255 now includes more cities and defines the zones encountered in large cities (30/255, October 1967)

1050 Teleguide now gives detailed assistance in designing this versatile device into your Teleprocessing system (70/1050, September 1968)

Airlines RPQ equipment is now described for possible use outside that industry (Sec. 75, May 1968)

Data Set descriptions and prices have been completely updated and now show where to obtain more technical reference material (20/12, July 1968)

Coming soon - Updated and expanded intrastate rates

A new modem and Data Set Configuration which guides both selection and timing of these important devices.

If your Handbook doesn't contain the issued material listed here you are not included in the controlled distribution. All DP Branch Office copies are registered to such titles as Branch Manager, Account Manager, SE Manager, Teleprocessing Coordinator, etc. Have your manager write to the Handbook Editor if an increase in quantity of revisions is needed. All other location copies of the Handbook are registered in individual name or title.

It is important to keep your Handbook up-to-date, know what is in it, and to use it, in order to save time and assure correctness.

Handbook Editor, Dept. H72
IBM Corporation
P. O. Box 390
Poughkeepsie, N. Y. 12602

SYSTEMS ENGINEERING COMMUNICATIONS

The wide spread installation of Field Engineering's 2740 leased line terminals in branches offers an improved SECOM/RETAIN retrieval capability. Accordingly, the following explanation is intended to summarize SECOM's objectives and to review the 2740 operating procedures.

SECOM is an information communication and retrieval system available to all branch offices, Field Systems Centers, Test Centers, and Datacenters. The system will provide new and timely information concerning IBM Type I and Type II Programming Systems on a daily basis.

SECOM contains information on 1130, 1800, and System/360 Programming Systems problems, solutions, changes, restrictions, and clarifications which are of general interest to Systems Engineering. SECOM, in combination with the FE RETAIN system which reports APAR's resulting in fixes to Type I problems, will help to avoid much of the duplication involved in solving common problems in different locations. The information in SECOM is developed from APAR submissions which will not result in fixes to programming systems, and from problems referred to the Headquarters and the District Field Systems Centers. SECOM also contains Type II APAR's and other information submitted through the Industry Development Centers. The SECOM input format (Branch Office Manual - page 4-955) may be used by the Branch Office to submit Type I or Type II information to SECOM through the appropriate Field Systems Center or Industry Development Center.

A new feature of SECOM is the inclusion of information on Type III programs under development. The abstracts for these programs are included in the weekly and monthly abstract listing and are identified by the prefix "prog". A given program's description will remain in SECOM until that program has been published in the Catalog of Programs.

Each morning abstracts of the information entered into SECOM during the previous day will be broadcast to each field location over ITPS.

Each week copies of the previous week's abstracts for SECOM and RETAIN, in sequence by program number, are mailed to each Branch Office. This weekly listing may be used to replace the file of daily ITPS listings. Each month copies of all current abstracts in the SECOM file are mailed to each branch. This monthly listing may be used to replace all other SECOM listings prior to that date.

All locations may inquire into the SECOM system to retrieve full text of an item by using their ITPS terminal, a dial-up 1050, or a leased line 2740. ITPS users are limited to a single text retrieval per request.

Direct use of the SECOM/RETAIN system is for IBM personnel only. The sign on procedure, user number, and retrieval operation must be restricted to IBM use only to insure the security of the system. Using your own good judgement, you may discuss appropriate information from SECOM/RETAIN with customers.

The 2740 communication procedures are:

1. The modem test switch, located on the left hand side of the terminal table must be set to the OFF position.

2. The mod switch called com-local on the right hand side of the keyboard must be set to the COM position.

3. The power on switch, located on the right hand side of the keyboard must be set to ON position.

Entry of data from a 2740 terminal is essentially a three step operation.

1. Depress the BID key and wait for the T light.
2. Type in data.
3. Depress EOB key

After the data connection has been established and the keyboard unlocked, you may sign on using the following procedure:

Entry sxxxx/retain (EOB)
(xxxx represents your registered SECOM number)

Response YOUR NAME IS Branch Office...
Enter RETAIN Request

This completes the sign on procedure. Specific information may now be obtained by following the message formats.

When all requests have been answered, sign off using the following procedure:

(Condition keyboard by depressing BID key)

Type sign off
Response LINE IS SIGNED OFF

When sign off has been accomplished, you may power off by depressing the power on-off switch to the off position.

For a detailed explanation of SECOM request formats and dial-up 1050 or ITPS operation, refer to the pocket size reference card entitled "SECOM/RETAIN Procedure--Revised 8/1/68" available through your SD&I Center.

SECOM is an early warning system for known problems and solutions on a daily basis. Use of this tool may save you valuable time and effort.

INFORMATION ABOUT THE NEWSLETTER

EVERY SALESMAN AND SYSTEMS ENGINEER SHOULD RECEIVE ONE COPY OF THE NEWSLETTER. **Each FE Branch Manager should receive five copies for distribution to Customer Engineers.**

The IBM Installation Newsletter is distributed only to IBM locations and is not available to customers. The Newsletter is separated into two parts. The first part contains information which is intended only for IBM personnel and the pages are labeled For IBM Internal Use Only. THESE PAGES, OR REPRODUCTIONS OF THEM, ARE NOT TO BE GIVEN TO CUSTOMERS. However, using your own good judgement, you may discuss pertinent information from these pages with a customer. The second part contains information which may be REPRODUCED by the Branch Office at their discretion and given to customers. These pages are not labeled with any restrictive classification. Note that when any abstracts for Type III programs are given to the customer, he must be informed that the disclaimer for Type III programs as contained in the Newsletter, applies. YOUR COOPERATION IN PROPERLY HANDLING THE CONTENT OF THE IBM INSTALLATION NEWSLETTER IS NECESSARY FOR ITS CONTINUED PUBLICATION.

Changes to the quantity of the Newsletter shipped are to be made by the Administration Manager. (NOTE THAT EXTRA COPIES OF THE NEWSLETTER ARE NOT TO BE ORDERED FOR DISTRIBUTION OF INFORMATION TO CUSTOMERS.) He should update and send the DP Administration Managers' Document Control card (Electro IBM N61508) to DP Document Control, DPD HQ.

Direct inquiries concerning the contents of this Newsletter to:

W. F. Malmborg, Editor
IBM Installation Newsletter
Dept. 812
DPD HQ
112 East Post Road
White Plains, New York 10601
Telephone 914 WH9-1900 Extension 4422
Tie Line 8-374- 4422

NEW PUBLICATIONS ANNOUNCED BY PRL'S

Information is extracted and condensed from the weekly PRL's (Publications Release Letters) to insure that all Salesmen and Systems Engineers are aware of new or revised Marketing Publications. Normally, each issue of the Newsletter will contain information extracted and condensed from two PRL's, one following the other. The information will be placed in the Newsletter in its original sequence with no rearrangement of form numbers or titles. It is not intended to replace existing information and distribution sources. You should be certain that you are aware of these sources.

NOTE:

Do not use the condensed Newsletter information to order publications. Most but not all (for example, some films and slide sets) of the items are available from Mechanicsburg. To receive the items you need without delay, prior to ordering check the REQ (Requisition From) code in the "Accumulative Supplement to IBM Publications Current Price List (Z20-0100)" which is attached to the weekly Publications Release Letter (PRL) and distributed to your Administration Manager. For an explanation of the REQ codes and other publications information see the first few pages of the "IBM Publications Current Price List (Z20-0100)".

The Publications Requisition, form number M02-0618 is used for ordering items from the IBM Distribution Center, Mechanicsburg, Pa.

(IBM World Trade Corporation has its own distribution center and procedures.)

PRL #46 November 15, 1968

FORM NO.	TITLE	DISPOSITION
N20-0030.22	Catalog of Programs for IBM System/360 - October 1968 Supplement to C20-1619-S	Scrap N20-0030.21
	MARKETING PUBLICATIONS	
320-1906-10	October 1968 In Brief	Scrap 320-1906-9
520-2100-0	Where real-time control is essential (IBM 1800)	NEW
540-0070-1	Carbon Ribbon Sample Box	Use 540-0078-0
543-0064-0	ET State & Local Catalog	NEW - Scrap 543-0045
A22-6813-4	IBM System/360 Model 40 Configurator	Scrap A22-6813-3, -2
A26-5847-S	IBM System/360 Model 20 Functional Characteristics	Use A26-5847-4, N33-1519 & N33-1521
A27-2731-1	IBM 2265 Display/2845 Display Control Component Description	Scrap A27-2731-0 & N27-2923
C28-2023-1	IBM System/360 Time Sharing System Master Index	Scrap C28-2023-0
C28-6396-0	IBM System/360 Operating System: USASI COBOL Language	NEW
C28-6535-3	IBM System/360 Operating System: Concepts and Facilities	Use C28-6535-2
H20-0492-1	IBM System/360 Decision Logic Translator (360A-CX-32X) Application Description Manual	Scrap H20-0492-0
H20-0523-0	IBM System/360 AD-APT/AUTOSPOT (OS) Numerical Control Processor	NEW
H20-0549-0	IBM System/360 AD-APT/AUTOSPOT (OS) Numerical Control Processor (360A-CN-12X) Part Programming Manual	NEW
H20-0557-0	IBM System/360 AD-APT/AUTOSPOT (OS) Numerical Control Processor (360A-CN-12X) Operations Manual	NEW
N20-1888	IBM 1130 Commercial Subroutine Package (1130-SE-25X) Version 2 Modification 1 Program Reference Manual Re: H20-0241-3	NEW
N28-2360	IBM System/360 Operating System: System Control Blocks Re: C28-6628-2	NEW
N28-2366	IBM System/360 Operating System: Messages and Codes Re: C28-6631-5	NEW
N28-2367	IBM System/360 Operating System: System Generation Re: C28-6554-4 with N28-2348 & N28-2325	NEW
Y20-0223-4	Sales Manual Inserts (programming only)	Scrap Y20-0223-3
Y20-0263-0	IBM S/360 Decision Logic Translator (360A-CX-32X) System Manual	NEW
Z20-1899-0	Construction and Civil Engineering Quick Reference Guide Facts Folder	NEW

PRL #47 November 22, 1968

FORM NO.	TITLE	DISPOSITION
N20-0360.52	IBM S/360 Newsletter Re: A22-6822-11	Scrap N20-0360.5
	MARKETING PUBLICATIONS	
505-0009-0	This is IBM	Abstract only
505-0010-0	A Computer Glossary	Abstract only
520-2100-0	Where real-time control is essential (IBM 1800)	Abstract only
540-0078-1	Carbon Ribbon Sample Box	Use 540-0078-0
A32-0007-0	2420 Model 5 Magnetic Tape Unit	NEW
C20-1689-1	An Introduction to the Compile-Time Facilities of PL/I	Use C20-1689-0
C33-4001-2	IBM S/360 Model 20 Input/Output Control System for the Binary Synchronous Communications Adapter	Scrap C33-4001-1
C33-4003-0	IBM S/360 Model 20 Remote Job Entry Work Station	NEW
C33-6006-1	IBM S/360 Model 20 Disk Programming System System Generation and Maintenance	Scrap C33-6006-0
E20-0118-0	Biomedical Application for an IBM Data Acquisition System	NEW
H20-0222-2	Project Control System/360 (360A-CP-06X) Version 2 Application Description	Scrap H20-0222-1
H20-0376-1	Project Control System/360 (360A-CP-06X) Version 2 Program Description and Operations Manual	Scrap H20-0376-0
H20-0564-0	IBM S/360 Matrix Language (MATLAN) (360A-CM-05X) Program Description Manual	NEW
N20-1885	IBM S/360 Document Processing System (360A-CX-12X) Program Description and Operations Manual Re: H20-0477-0	NEW
N20-1894	TNL for Advanced Life Information System Operations Manual Re: H20-0517-0 with N20-1875	NEW
N20-1895	TNL for ALIS Utility Program Description Manual Re: H20-0519-0 with N20-1876	NEW
N24-0424	TNL to IBM S/360 Model 25 (2540 Emulation in the 2560) Re: A24-3524-0	NEW
N28-3045	IBM S/360 Time Sharing System FORTRAN IV Re: C28-2007	NEW
N28-3047	IBM S/360 Time Sharing System Assembler Language Re: C28-2000	NEW
N33-8020	IBM 1800 Time-Sharing Executive System, Concepts and Techniques Re: C26-3703-0	NEW
R20-1078-0	IBM S/360 Operator Reference Guide	NEW
Y20-0284	TNL for ALIS - File Maintenance Premium Due, Overdue, and Anniversary Processing Transactions Re: Y20-0194-0	NEW
Y20-0291	TNL for ALIS - Accounting Control Program System Manual Re: Y20-0201-0 with Y20-0245	NEW
Y28-3095	IBM S/360 Time Sharing System System Service Routine PLM Re: Y28-2018	NEW

PRL #48 November 29, 1968

FORM NO.	TITLE	DISPOSITION
A26-3565-5	IBM S/360 Model 20 Bibliography	Scrap A26-3565-4
	MARKETING PUBLICATIONS	
221-0414-3	The IBM 1130 Computing System	Scrap 221-0414-2
520-2123-0	IBM 2980 for Savings and Loans	NEW
A22-6846-3	IBM 2702 Transmission Control: Component Description	Scrap A22-6846-2 & N27-3026
C28-6538-6	IBM S/360 Operating System: Linkage Editor	Use C28-6538-5 & N28-2330
C28-6644-2	IBM S/360 Operating System: Master Index	Scrap C28-6644-1
C33-2003-0	IBM S/360 Conversion Aids: The 1410/7010 Simulator for IBM S/360	NEW
H20-0252-3	IBM 1130 Scientific Subroutine Package (1130-CM-02X) Programmer's Manual	Scrap H20-0252-2
H20-0297-2	IBM S/360 Administrative Terminal System - OS (ATS/OS) Application Description Manual	Scrap H20-0297-1
H20-0533-1	Shared Hospital Accounting System (SHAS) (360A-UH-11X) Program Description Manual	Scrap H20-0533-0 & N20-1099
H20-0548-0	IBM S/360 Model 20 Hospital Patient Billing Application Description Manual	NEW
K20-0272-0	IBM S/360 Model 20 at Coburg Dairy, Inc.	NEW
N20-1892	TNL for Alis - Application Description Re: H20-0126-3	NEW
N20-1893	TNL for Advanced Life Information System Policy Master Record Code Book Re: H20-0483-0 with N20-1096	NEW
N20-1899	TNL to 1130 Statistical System (1130-CA-06X) User's Manual Re: H20-0333-1 with N20-1059	NEW
N20-1900	TNL to Structural Engineering System Solver (STRESS) for the IBM 1130 (1130-EC-03X) Version 2, User's Manual Re: H20-0340-2	NEW
N20-1903	TNL to STRAM DOS/360 User's Manual Re: H20-0346-0	NEW
N20-1908	TNL for Retail IMPACT Fashion System under OS/360 (360A-DR-04X) Program Description Man. Re: H20-0480-0 with N20-1084	NEW
N20-1909	TNL for Retail IMPACT Fashion System under DOS/360 (360A-DR-08X) Program Description Re: H20-0540-0	NEW
N20-1910	TNL for Retail IMPACT Fashion System under DOS/360 (360A-DR-08X) Operations Manual Re: H20-0541-0	NEW

For IBM Internal Use Only

PRL #48 (Continued)

N20-1911	TNL for Retail IMPACT - Inventory Management Program and Control Techniques - Application Description Re: E20-0188-2 and -3 with N20-1068 to -2 only	NEW
N20-1912	TNL to General Purpose Simulation System/360 User's Manual Re: H20-0326-2 with N20-1100	NEW
N20-1913	TNL to General Purpose Simulation System/360 Introductory User's Manual Re: H20-0304-3, -2 -1, -0 to -2, N20-1878, to -1 and -0 N20-1087 & N20-1067	NEW
N21-0097	IBM 1287 SRL TNL Re: A21-9064-2 with N21-0093	NEW
N27-2920	IBM S/360 Model 67 Functional Characteristics TNL to A27-2719-0	NEW
N33-8021	IBM 1130/1800 Basic FORTRAN IV Language Re: C26-3715-3	NEW
R20-8076-0	Basic Punched Card Procedures Education Guide	Scrap X24-3083-3
X24-3083-4	1440 Physical Planning Template	NEW
Y20-0222-5	Inserts (less programming)	Scrap Y20-0223-4
Y20-0223-5	Sales Manual Inserts 9programming only)	NEW
Y20-0282-0	TNL for ALIS - File Maintenance Status Transactions - Process and Quote Miscel., Loan Values, and Particip. Values System Manual Re: Y20-0192-0 with Y20-0242	NEW
Y20-0313	TNL - Service for Consultants Update Re: Y20-0161-1	NEW
Y27-7183	IBM 1130/2250 Graphic Subroutine Package for Basic FORTRAN IV Re: Y27-7174	
Z77-8065-1	DOS BTAM Coding for the Leased Line IBM 2780 Data Transmission Terminal	Z77-8065-0
Z77-8120-0	1030 Time and Attendance System	NEW
Z77-8121-0	Using 360 GPSS to Simulate the Operation of a Typical Non-Assembly Line Manufacturing Facility	NEW
Z77-8122-0	Budgetary Accounting on Small Direct Access Systems	NEW
Z77-8123-0	CIF Name Key Design	NEW
Z77-8124-0	CIF Preparation Guide	NEW
Z77-8127-0	Simulation of a Teleprocessing System for a Swedish Bank Using the S/360 Computer System Simulator	NEW
Z77-8128-0	A Real-Time Approach to the On-Line Data Entry Application	NEW
Z77-8129-0	Faster Quick Reference Manual for Programmers	NEW

December 13, 1968 Issue No. 68-24

IBM INSTALLATION NEWSLETTER

TABLE OF CONTENTS

OS/360 SER 1 OPTION, ONE SYSGEN FOR TWO SYSTEMS CLARIFICATION

Installation Newsletter 68-19 contained an article on this subject in which "INCLUDE" was used in the general sense to point to the proper module. This may not have been clear, for a reader has pointed out that the actual JCL statements should be:

```
INCLUDE  MODLIB (IFBSR150)  for Model 50
INCLUDE  MODLIB (IFBSR140)  for Model 40
```

The above applies to a PCP or MFT system. For an MVT System, these statements should read:

```
INCLUDE  MODLIB (IFBSR350)  for Model 50
INCLUDE  MODLIB (IFBSR340)  for Model 40
```

This method will neither apply nor be necessary with OS Release 17 because of the splitting of the MODLIB and the inclusion of Model Independent SER. These new features are described in Installation Newsletter 68-23.

OS/360 MULTIPLE "CLG'S" PER STEP CORRECTION

When more than one Compile Load-and-go Procedure is executed in a Job Step where the Syslmod DD card has a disposition of (MOD, Pass) and a member name is specified in the DD card, then the STOW is performed with the "ADD" function rather than the "REPLACE" function. This prevents more than one "CLG" per Job Step.

This can be corrected by a SUPERZAP fix:

NAME	IEWL	IEWL	MFNL
VERIFY	016A	4780	C170
REP	016A	4700	C170

OS/360 JOB PLANNING

To assist the user in the successful planning of job descriptions and/or datasets the following considerations apply.

1. A maximum of 248 I/O device addresses can be included in the system. Reference Installation Newsletter 68-21. Special techniques can be used to support more than 248 addresses.
2. A job can include a maximum of 255 separate steps.
3. A job step can include a maximum of 255 DD statements.
4. A job step can request a maximum of 683 volume serial numbers.
5. A single DD statement can request a maximum of 59 units.
6. A single DD statement can request a maximum of 255 volume serial numbers.

OS/360 RELEASE 15/16 CHANGING THE WAIT TIME LIMIT FOR JOB STEP TIMING

This contribution has not been submitted to any formal IBM test. Potential users should evaluate its usefulness in their own environment prior to implementation.

The Job Step Timing feature (SYSGEN option) in Release 15/16 terminates a job if it has waited on an event for 30 minutes (ABEND code 522). If an installation requires job step timing and either has jobs which could wait on an event(s) for more than 30 minutes or desires the termination of a job after a wait of less than 30 minutes, the installation can alter the wait time limit to conform to its requirements.

To compute the wait time limit multiply the hexadecimal value '0023366C' by the number of minutes contained in the wait time limit you intend to implement. For example, a 10-minute wait time limit would equal '01602038' (hexadecimal). The computed hexadecimal value is to be placed on the REP card in the Superzap deck in place of 'xxxxxxxx'. See Exhibit 1.

Because the update is applied to the nucleus as it resides on direct access, the system should be re-IPL'ed after the update is made.

OS/360 RELEASE 15/16 CHANGING THE WAIT TIME LIMIT FOR JOB STEP TIMING

```
//JOBCRD      JOB      ,,MSGLEVEL=1
//STEP        EXEC     PGM=SUPERZAP
//SYSLIB      DD       DSNAME=SYS1.NUCLEUS,UNIT=2311,DISP=OLD,
//                     VOLUME=SER=yyyyyy
//SYSPRINT    DD       SYSOUT=A
//SYSIN       DD       *
  NAME     IEANUC01   IGC001
  VERIFY     03E4     042060B1
  REP        03E4     xxxxxxxx
/*
```

EXHIBIT 1

OS/360 MVT SYSTEMS QUE SPACE

In analyzing an MVT core dump, it sometimes becomes necessary to determine the location of SQS, or the amount of SQS being used.

The MVT supervisor PLM states that GOVERFLB is a three word control block containing an address pointer to SQS. It does not, however, explain how to find GOVERFLB in a core dump. The following is one method that could be used.

GOVERFLB - This is a three word control block that contains a pointer for SQS space and a pointer for free dynamic core not assigned to any region, and a pointer to the first address above SQS. (See MVT Supervisor PLM Y28-6659). It will probably have a different address in each release. The control block is actually located within the GETMAIN Module and can be found with the help of a SYSABEND or stand alone core dump and the Stage II assembly of the NUCLEUS:

1. Get the address of the NEW SVC FLIH pointer from core dump location 60 (Hex).
2. To this address, add the displacement of SVCCONS + 4. This is found in the assembly of the SVC FLIH in the NUCLEUS. The result is a pointer to the Resident SVC table.
3. Find the SVC 4 pointer in the table (SVCTABLE + 16).
4. To that address, add the displacement of GOVERFLB from the beginning of the SVC 4 GETMAIN Assembly (IGC004) in the NUCLEUS. The result is the location of GOVERFLB.
5. Check the format of GOVERFLB in the MVT Supervisor PLM. The second word is a pointer to the Descriptor Que Element for SQS, which gives the beginning of SQS, the length of SQS, and the beginning of the Free Que chain for SQS.

OS/360 PL/I ASSEMBLER LANGUAGE ROUTINE TO DETERMINE CORE AVAILABLE IN A MVT REGION

The following contribution has not been submitted to any formal IBM test. Potential users should evaluate its usefulness in their own environment prior to implementation.

The program is an assembler routine callable from PL/I that returns the amount of core currently available within the Region. This information can be useful in determining optimum region size.

The qunatity returned in the sum of:

1. The amount not allocated to PL/I.
2. The amount allocated to PL/I but not yet used.

This routine has been checked out on MVT Release 15/16.

OS/360 PL/1 ASSEMBLER LANGUAGE ROUTINE TO DETERMINE CORE AVAILABLE IN A MVT REGION

```
*    ASSEMBLER ROUTINE TO COUNT ALL FREE CORE
*     TO USE FROM PL/I - DCL SPACE BINARY FIXED(31,0)
*                        CALL MAPMVT (SPACE)
*   AMOUNT CAN BE PRINTED    PUT EDIT ('FREE',SPACE0(A,F(31,0))
*
         MAPMVT  CSECT
         IHEQLWO DXD     A
         IHEQSFC DXD     A
                 STM     14,12,12(13)
                 L       11,*-*(12)
                 ORG     *-2
                 DC      QL2(IHEQLWO)
                 ST      13,4(0,11)
                 LR      13,11
                 BALR    10,0
                 USING   *,10
                 USING   MYWORK,11
                 LR      2,1
                 SR      3,3             ZERO ACCUMULATOR
          * ARBITRARY USE OF SUBPOOL 9
          MO     GETMAIN  VC,LA=MINMAX,A=ADDRQTY,SP=9,MF=(E,DUMGET)
                 LTR     15,15
       1         BNZ     FIN
                 A       3,ADDRQTY+4     FREE QUANTITY
                 B       MO
          FIN    FREEMAIN R,SP=9
                 L       5,*-*(12)
                 ORG     *-2
                 DC      QL2(IHEQSFC)
          TEST   LTR     5,5
       2         BZ      FINIS           END OF CHAIN
                 A       3,0(0,5)        FREE CORE AMT
                 L       5,12(0,5)       CHAIN FWD PTR
                 B       TEST
         FINIS   L       4,0(2)
                 ST      3,0(4)
                 L       13,4(0,13)
                 LM      14,12,12(13)
                 BR      14
         MINMAX  DC      F'8'
                 DC      X'00FFFFF0'
         MYWORK  DSECT
                 DS      9D
         ADDRQTY DS      2F
         DUMGET  GETMAIN  VC,LA=0,A=0,SP=9,MF=L
                 END
```

EXHIBIT 1

OS/360 PL/I F INCOMPATIBLE CHANGES IN RELEASE 16

Incompatible changes in programs produced by the fourth version of the PL/I F compiler have been documented in the PL/I (F) Programmer's Guide (C28-6594-3, N33-6007) pages 11-12.1.

The following items should be added to that list:

1. STOP, EXIT, CALL IHEDUMP and termination by standard system action no longer produce a system ABEND, although the condition code remains the same, thus subsequent job steps will be executed.

Appropriate COND parameters in later steps may be used to suppress execution. (The PL/I (F) Reference Manual and Programmer's Guide will be updated to reflect this change).

2. The FINISH condition will no longer terminate execution of a PL/I program. Both standard system action and normal return are to continue with the next sequential statement (See PL/I (F) Reference Manual C28-8201-1 page 268).

3. The high-order bit of the last word in a parameter list is now being flagged (i. e. bit = '1'B). This may affect the execution of some assembly language subroutines.

Official specifications will be as provided through the formal IBM procedures.

OS/360 RELEASE 15/16 2250 GRAPHICS WITH PL/I CORRECTION

The following item is a correction to the article OS/360 Release 15/16 2250 Graphics with PL/I which appeared in Installation Newsletter 68-17. The change is to the coding in Exhibit 1 to allow correct return of results for those functions returning decimal float values.

Beginning with OS - Release 15/16, PL/I will insert a hex '80' character in the high order byte of the last parameter of argument lists being passed by CALL statements to subroutines. This agrees with the convention already used by FORTRAN and COBOL. It allows variable length argument lists to be passed to subroutines which utilize the feature.

GSP (Graphic Subroutine Package) does utilize this feature, and with its addition to PL/I, GSP Subroutine subprograms can now be called directly from PL/I. No assembler language interface is required.

Because of differences in the way results are returned, the GSP Function subprograms still cannot be used directly by PL/I. The assembler language interface shown in Exhibit 1, however, will allow PL/I to utilize these GSP functions. Notice that this interface has the two CSECTS IGSP and RGSP. IGSP is used for those functions returning a Binary Fixed Value and RGSP is used for those functions returning a Decimal Float value.

The following example shows how the GSP function ITST would be used.

```
ICODE = IGSP (ITST, IGDS, I);
```

IGDS and I are the arguments used by ITST. Upon return from the function, ICODE will contain a Binary Fixed value.

Additional GSP information can be found in the publication, Graphic Programming Services for FORTRAN IV, form number C27-6932.

See Exhibit page 6.

OS/360 FORTRAN RE-READ ROUTINE-MODIFICATION FOR RELEASE 15/16

Issue 67-09 of the Installation Newsletter contained on Page 13 a program listing of a FORTRAN routine called READRE. This routine is used by many OS/360 FORTRAN users to allow the FORTRAN "REREAD" capability.

Release 15/16 contained a new object time error handling routine that has changed the FORTRAN library linkages. Consequently, the READRE routine as originally published will not work with Release 15/16. Following is a listing of the routine showing modifications needed to run with Release 15/16. This routine has not been submitted to any formal IBM test.

See Exhibit page 6.

OS/360 FORTRAN CORE ROUTINE-MODIFICATION FOR RELEASE 15/16

The FORTRAN CORE read/write routine first appeared in the Installation Newsletter in Issue No. 66-12. It performs a function similar to the RE-READ routine described above, in allowing the user to re-scan a buffer after initial read-in. CORE needs a single modification for use with OS Release 15/16 and that change is included in the following listing. Note the change at CORE0860. The old instruction remains as a comment card. This routine has not been submitted to any formal IBM test.

See Exhibit pages 7 and 8.

OS/360 RELEASE 15/16 2250 GRAPHICS WITH PL/I CORRECTION

```
RGSP     CSECT
         USING *,15
         BC    15,BBB
         DC    X'5'
         DC    CL5'RGSP '
BBB      STM   14,12,12(13)
         DROP  15
         BALR  3,0
          USING *,3
         LR    10,13
         LA    13,ARA
         ST    13,8(10)
         ST    10,4(13)
         L     5,0(1)
         L     15,0(5)
         LA    1,4(1)
         BALR  14,15
         S     1,DD8
         STE   0,0(1)
         L     13,ARA+4
         LM    2,12,28(13)
         L     14,12(13)
         MVI   12(13),X'FF'
         BCR   15,14
ARA      DS    18F
DD8      DC    F'8'
IGSP     CSECT
         USING *,15
         BC    15,AAA
         DC    X'5'
         DC    CL5'IGSP '
AAA      STM   14,12,12(13)
         DROP  15
         BALR  3,0
          USING *,3
         LR    10,13
         LA    13,ARE
         ST    13,8(10)
         ST    10,4(13)
         L     5,0(1)
         L     15,0(5)
         LA    1,4(1)
         BALR  14,15
         S     1,D8
         ST    0,0(1)
         L     13,ARE+4
         LM    2,12,28(13)
         L     14,12(13)
         MVI   12(13),X'FF'
         BCR   15,14
ARE      DS    18F
D8       DC    F'8'
         END
```

EXHIBIT I

OS/360 FORTRAN RE-READ ROUTINE

```
STMT   SOURCE STATEMENT
   2        PUNCH  'READRE'
   3             START
   4             ENTRY READRE
   5             EXTRN IBCOM#
   6             EXTRN FIOCS#
   7 READRE      LR    8,15
   8             USING READRE,8
   9             ST    1,SV1
  10             L     1,ADZZZ
  11             L     15,ADIBCOM
  12             MVI   74(15),X'50'
  13             EX    0,74(15)
  14             MVI   74(15),X'58'
  15             L     1,SV1
  16             BCR   15,14
  17             USING *,1
  18 ZZZ         STM   4.15,SAVE
  19             DROP  1
  20             LR    11,1
  21             USING ZZZ,11
  22             LR    4,0
  23             CLC   0(2,4),H00F0
  24             BC    8,CHK99
  25             L     1,ADFIOCS
  26             LM    4,15,SAVE
  27             BCR   15,1
  28 CHK99       CLC   0(4,2),I99
  29             BC    8,BYPASS
  30             L     1,ADFIOCS
  31             BALR  0,1
  32             DC    X'00F0'
  33             B     ERROR
  34             ST    2,SV1
  35             ST    3,SV2
  36             BC    15,LEAVE
  37 BYPASS      L     2,SV1
  38             L     3,SV2
  39 LEAVE       LA    1,6(4)
  40 EXIT        LM    4,15,SAVE
  41             BCR   15,1
  42 ERROR       LA    1,2(4)
  43             B     EXIT
  44 ADZZZ       DC    A(ZZZ)
  45 ADIBCOM     DC    A(IBCOM#)
  46 SAVE        DC    18F'0'
  47 ADFIOCS     DC    A(FIOCS#)
  48 I99         DC    F'99'
  49 SV1         DS    F
  50 SV2         DS    F
  51 H00F0       DC    X'00F0'
  52             END
```

*Note changes in lines 5, 6, 45, 47

EXHIBIT 1

OS/360 FORTRAN CORE ROUTINE--MODIFICATION FOR RELEASE 15/16

```
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * CORE0010
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * CORE0020
*                                                                                       * CORE0030
*                          CORE READ WRITE ROUTINE                                      * CORE0040
*                                                                                       * CORE0050
*                                                                                       * CORE0060
* THIS SUBROUTINE MODIFIES IBCOM IN CORE IN SUCH A WAY THAT THE NEXT                    * CORE0070
*READ OR WRITE REQUEST WILL CAUSE THE INFORMATION IN THE USER'S                         * CORE0080
*BUFFER TO BE CONVERTED AND PLACED IN THE USER'S LIST OR THE INFORMA-                   * CORE0090
*TION IN THE USER'S LIST TO BE CONVERTED AND PLACED IN THE BUFFER                       * CORE0100
*                                                                                       * CORE0110
*                          CALL CORE(A,N)                                               * CORE0120
*                                                                                       * CORE0130
* WHERE A IS THE NAME OF AN ARRAY SUFFICIENTLY LARGE TO CONTAIN THE                     * CORE0140
* INFORMATION TO BE PROCESSED - A IS A SIMULATED BUFFER - AND N IS                      * CORE0150
* THE BUFFER LENGTH IN BYTES.                                                           * CORE0160
*                                                                                       * CORE0170
* EXAMPLE                                                                               * CORE0180
*                                                                                       * CORE0190
*         DIMENSION A(20), WD(80)                                                       * CORE0200
*         READ (5,100) WD                                                               * CORE0210
*         CALL CORE (A,80)                                                              * CORE0220
*         WRITE (10,100) WD                                                             * CORE0230
*    100  FORMAT (80A1)                                                                 * CORE0240
*         CALL CORE (A,80)                                                              * CORE0250
*         READ (10,101) X, Y, Z                                                         * CORE0260
*    101  FORMAT (3F8.5)                                                               * CORE0270
*                                                                                       * CORE0280
* A CALL TO CORE MUST PRECEDE EACH DUMMY READ OR WRITE.                                 * CORE0290
* IT IS POSSIBLE TO DO DOUBLE OR MULTIPLE ARRAYS AND USING THE                          * CORE0300
*    APPROPRIATE ARRAY NAME IN THE CALL TO CORE.                                        * CORE0310
*         OS/360 RELEASE 15/16 CORRECTION AT CORE0860                                   CORE0315
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * CORE0320
CORE#    START 0                                CORE=                                 CORE0330
         ENTRY CORE                                                                    CORE0340
         EXTRN IBCOM#                                                                  CORE0350
         EXTRN FIOCS#                                                                  CORE0360
*                                                                                       * CORE0370
*              FIRST ENTRY POINT.  THIS ROUTINE PICKS UP OPERANDS ONE                   * CORE0380
*              AND TWO AND STORES THEM AT THE TWO FULL WORDS AT BUFLOC                  * CORE0390
*                                                                                       * CORE0400
         USING *,15                                                                    CORE0410
CORE     B     *+12              BRANCH AROUND NAME                                    CORE0420
         DC    XL4'07000000'     ROUTINE NAME FOR CALL TRACE                           CORE0430
         DC    CL4'CORE'         ROUTINE NAME FOR CALL TRACE                           CORE0440
         STM   14,3,12(13)       SAVE REGISTERS                                        CORE0450
         LM    2,3,0(1)          FETCH OPERAND ADDRESSES                               CORE0460
         L     3,0(3)            FETCH OPERAND 2 (LENGTH)                              CORE0470
         STM   2,3,BUFADR        STORE BUFFER LOC AND LENGTH                           CORE0480
         LA    1,CORE2           R1=A(SECOND ENTRY POINT)                              CORE0490
         LA    3,CLOAD           SET BASE REGISTER FOR CLOAD                           CORE0500
         BALR  2,3               LINK TO MODIGY IBCOM ADCON                            CORE0510
         LM    14,3,12(13)       RESTORE REGISTERS                                     CORE0520
         SR    15,15             SUPPRESS VARIABLE RETURN                              CORE0530
         BR    14                RETURN                                                CORE0540
```

EXHIBIT 1

OS/360 FORTRAN CORE ROUTINE -MODIFICATION FOR RELEASE 15/16

```
         DROP  15                                                       CORE0550
         EJECT                                                          CORE0560
*                SECOND ENTRY POINT. IBCOM ENTERS AT CORE2 THINKING IT  * CORE0570
*                WENT TO FIOCS. THIS ROUTINE SIMULATES FIOCS BY POINTING* CORE0580
*                TO BUFFER ADDRESS AND LENGTH STORED BY FIRST ROUTINE.  * CORE0590
*                IBCOM IS RESTORED TO NORMAL, FOLLOWED BY RETURN TO     * CORE0600
*                IBCOM.  A WRITE BUFFER IS INITALIZED TO BLANKS BEFORE  * CORE0610
*                IBCOM FILLS IT TO ALLOW T FORMAT TO WORK CORRECTLY.    * CORE0620
*                                                                       * CORE0630
         USING *,1                                                      CORE0640
CORE2    ST    4,SAVE4          SAVE R4.                                CORE0650
         LR    4,1              R4=A(CORE2)                             CORE0660
         USING CORE2,4                                                  CORE0670
         DROP  1                                                        CORE0680
         LR    1,0              R1 POINTS TO FIOCS CALL PARAMETERS      CORE0690
         TM    1(1),X'0F'       TEST FOR OUTPUT, FIRST TIME             CORE0700
         BO    OUTPUT           BRANCH TO FIRST OUTPUT ROUTINE          CORE0710
         L     1,VFIOCS         R1=A(FIOCS) TO RESTORE IBCOM            CORE0720
         LA    3,CLOAD          SET BASE REGISTER FOR CLOAD             CORE0730
         BALR  2,3              LINK TO MODIFY IBCOM ADCON              CORE0740
         LM    2,3,BUFADR       LOAD ARRAY ADR AND LENGTH               CORE0750
         B     RETURN           BRANCH TO RETURN TO IBCOM               CORE0760
OUTPUT   LM    2,3,BUFADR       LOAD ARRAY ADR AND LENGTH               CORE0770
         MVI   0(2),X'40'       BLANK FIRST BUFFER LOCATION             CORE0780
         BCTR  3,0              -1 L-1 CHAR TO BE BLANKED               CORE0790
         BCTR  3,0              -1 LENGTH CODE FOR MOVE=LENGTH-1        CORE0800
         EX    3,DMOVE          EXECUTE DUMMY MOVE TO CLEAR BUFFER      CORE0810
         LA    3,2(3)           R3=R3+2 RESTORE ORIGINAL LENGTH         CORE0820
RETURN   L     4,SAVE4          RESTORE R4                              CORE0830
         LR    1,0              R1=A(IBCOM ARGUMENTS)                   CORE0840
         DROP  4                                                        CORE0850
*        B     2(1)             RETURN TO IBCOM.  REL. 14 & PRIOR.      CORE0860
         B     6(1)             RETURN TO IBCOM, REL. 15/16 & SUBSEQUENT. CORE0860
         SPACE 2                                                        CORE0865
DMOVE    MVC   1(0,2),0(2)      EXECUTED. CLEARS UP TO 257 BYTE BUFFER  CORE0870
         EJECT                                                          CORE0880
*                R1= AN ADDRESS THE CALLER WANTS STORED AT VFIOCS IN    * CORE0890
*                R15 MUST BE A(IBCOM) TO SATISFY BASE REG  REQMNTS IN   * CORE0900
*                IBCOM. CALLER LOADS R3=A(CLOAD) FOR ME                 * CORE0910
*                                                                       * CORE0920
         USING *,3                                                      CORE0930
CLOAD    ST    15,SAVE          SAVE R15                                CORE0940
         L     15,VIBCOM        R15=A(IBCOM) FOR IBCOM BASE REG         CORE0950
         MVI   74(15),X'50'     MAKE  LOAD  A STORE INSTRUCTION         CORE0960
         EX    0,74(15)         STORES R1 AT VFIOCS IN IBCOM            CORE0970
         MVI   74(15),X'58'     RESTORE LOAD INSTRUCTION TO STORE       CORE0980
         L     15,SAVE          RESTORE R15                             CORE0990
         BR    2                RETURN                                  CORE1000
BUFADR   DS    2F               STORAGE FOR A(BUFFER) AND ITS LENGTH    CORE1010
SAVE     DS    F                STORAGE FOR R15                         CORE1020
SAVE4    DS    F                STORAGE FOR R4                          CORE1030
VIBCOM   DC    A(IBCOM#)        A(L 1, VIBCOM INSTN IN IBCOM-74)        CORE1040
VFIOCS   DC    A(FIOCS#)        ADDRESS OF FIOCS ROUTINE                CORE1050
         END                                                            CORE9999
                                  CARD COUNT      109
```

EXHIBIT 1

OS/360 ISAM DATA SET INTEGRITY

OS/360 users can prevent the loss of ISAM Data Set records when a system ABEND occurs during a record addition that moves a record from the Prime Area to the Overflow Area. The DCB parameter of WMSA causes the in core adjustment of the primary and the overflow records before the writing occurs. Thus, the prime area is not over written before the moved record is set up in the overflow area. This technique requires more core but eliminates the exposure to lost records in this situation.

OS/360 USE OF ACCOUNTING ROUTINES WITH MFT-II AND MVT

The Master Scheduler utilizes Reader/Interpreter and Initiator modules to start a Reader or Writer and, in MVT, an Initiator. As a result of a START command, the system provides a JOB statement, and the remaining JCL is taken from PROCLIB. This means that your accounting routine may be looking for accounting information which the system does not provide. This would lead you to fail the "job" in your accounting routine, when in reality it is a System Task.

A solution is to check the system provided jobname in your accounting routine and not fail selected names based on the lack of accounting data. The procedure name used in the START command becomes the system generated jobname.

DOS/360 INITIALIZE DISK-MODIFICATIONS FOR FG USE

The following field contribution has not been submitted to any formal IBM test. Potential users should evaluate it in their own environment prior to implementation.

The coding sheets in Exhibit 1 show the procedure and modifications needed to catalog the DOS initialize disk program to operate in F2.

Punch out the IJWID modules from the relocatable library. Change phase names for cataloging for F2. Include the three Rep cards that change the fetch phase names in modules IJWID and IJWID2. Catalog the modules to core image library. The same procedure can be used for F1.

DOS/360 PL/I CLARIFICATIONS

Issue No. 68-17 of the Newsletter contained an article under this same title. Paragraph No. 3 on "Internal Representation of Pointers" may have been misleading, and the following information has been supplied for additional clarification.

It is true that pointers presently occupy 3 bytes, however, these 3 bytes are the rightmost 3 bytes of a full word. The first byte of the full word is normally unused. However, within structures the data item ahead of the pointers may extend into the first byte of the pointer full word. Pointers are not generally used as elements of structures since structures are normally used as input or output records, and pointer values, specifying an absolute address, are meaningful only during the execution of a particular program.

Four byte, full word, pointers are planned for PL/I-D Version 4, DOS Release 20, and it is expected that only a very few programs will be effected by this change.

IBM System 360 Assembler Coding Form

PROGRAM MODIFICATIONS TO INITIALIZE DISK FOR FG | PUNCHING INSTRUCTIONS | GRAPHIC | PUNCH

PROGRAMMER | DATE

Name | Operation | Operand | Comments

```
/ JOB PUNCH
/ EXEC RSERV

     PUNCH OUT THESE MODULES FROM REL
          IJWID
          IJWID1
          IJWID2
          IJWID3
          IJWID4

JWID
PHASE INTD,*,NOAUTO          'CHANGE PHASE NAME EX. PHASE INTDA,*,NC
INCLUDE IJJCPDIN
~~INCLUDE IJWID1~~           'ELIMINATE'
PHASE INTD2,IJWID1P1,NOAUTO  'CHANGE PHASE NAME EX. PHASE INTDB,IJW1
~~INCLUDE IJWID2~~           'ELIMINATE'
PHASE INTD3,IJWID2P2,NOAUTO  'CHANGE PHASE NAME EX. PHASE INTDC,IJW1
~~INCLUDE IJWID3~~           'ELIMINATE'
PHASE INTD4,IJWID1P1,NOAUTO  'CHANGE PHASE NAME EX. PHASE INTDD,IJW1
~~INCLUDE IJWID4~~           'ELIMINATE'
          END
```

EXHIBIT 1

DOS/360 INITIALIZE DISK-MODIFICATIONS FOR FG USE

IBM System 360 Assembler Coding Form

```
                    'PLACE PHASE CARDS AHEAD OF EACH RESPECTIVE MODULE
                       AND CATALOG.'
                    'THE INCLUDE IJJCPDIN CARD GOES IN 1ST MODULE BEHIND
                       PHASE CARD, OTHER INCLUDE CARDS ARE ELIMINATED.

// JOB CATAL
// OPTION CATAL
   INCLUDE
   ACTION F2          'TO CATALOG TO F2'
   PHASE INTDA,*,NOAUTO
   INCLUDE IJJCPDIN
      (MODULE IJWID1 WITH REP CARD)
   PHASE INTDB,IJWID1P1
      (MODULE IJWID2 WITH 2 REP CARDS)
   PHASE INTDC,IJWID2P2,NOAUTO
      (MODULE IJWID3  NO REP CARDS)
   PHASE INTDD,IJWID1P1,NOAUTO
      (MODULE IJWID4  NO REP CARDS)
/*
// EXEC LNKEDT
/&
                    'PROGRAM RETRIEVED BY // EXEC INTDA'
                    'YOU STILL CATALOG BG INTDSK AS USUAL'
```

EXHIBIT 1 (Continued)

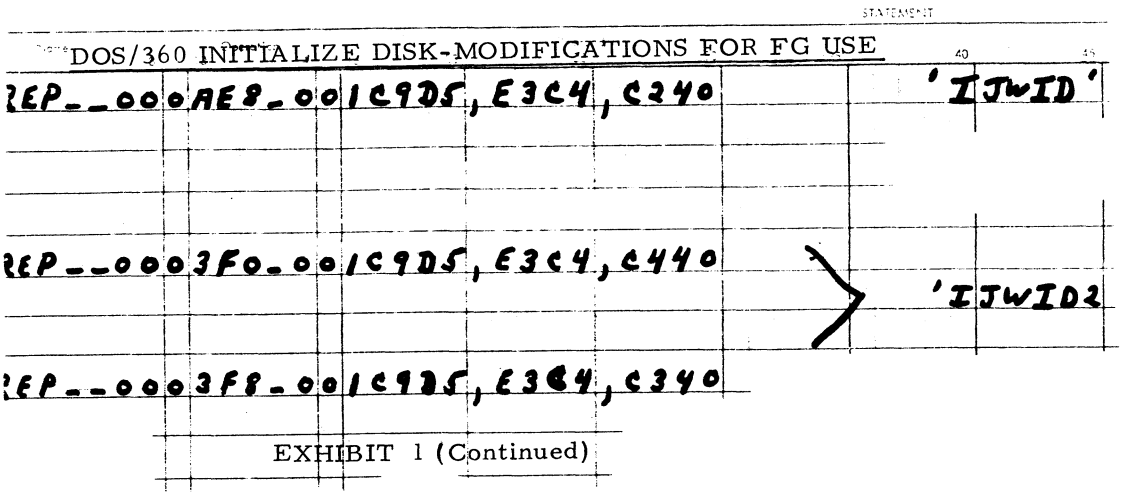


EXHIBIT 1 (Continued)

NEW TYPE III PROGRAMS

The following are the abstracts of programs which have been recently made available from the Type III library.

The program, along with its complete abstract, will be incorporated in subsequent issues of the Catalog of Programs.

Programs may be obtained by submitting a properly completed "General Program Request Card" (Form Number 120-1145) to the Program Information Department, 40 Saw Mill River Road, Hawthorne, New York, 10532.

These programs and their related documentation are distributed by IBM in the author's original form and have not been subjected to any formal testing.

Any discussions of Type III programs with customers must emphasize the following points:

1. - Type III (IBM employee-contributed) programs are provided by the IBM Corporation as part of its service to customers.

2. - Type III programs have not been subjected to any formal test.

3. - Recipients of Type III programs are expected to make the final evaluation as to the usefulness of the programs in their own environment.

4. - There is no committed maintenance for Type III programs. However, any changes the author chooses to make will be announced in subsequent issues of the Catalog of Programs.

5. - IBM makes no warranty, expressed or implied, as to the documentation, function, or performance of Type III programs.

* * * * * * * * * * *

\- - - REVISION - - -

DOS PRIVATE LIBRARY

The Machine Readable Material has been changed.

Ordering Procedure: Program Order Number 360D-03.1.012

The program material can be obtained in card form or on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

If not specified, a 9 track DTR will be provided. No tape submittal is required. The DTR will be provided by the library.

* * * * * * * * * * *

\- - - REVISION - - -

BASIC AUTOCODER ACCESS METHOD

14 pages of Documentation and the Machine Readable Material have been changed.

Ordering Procedure: Program Order Number 360D-06.3.003

The program material can be obtained in card form or on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

If not specified, a 9 track DTR will be provided. No tape submittal is required. The DTR will be provided by the library.

* * * * * * * * * * *

\- - - REVISION - - -

ESCAP

25 pages of Documentation and the Machine Readable Material have been changed.

Ordering Procedure: Program Order Number 360D-12.2.006

The program material can be obtained on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

If not specified, a 9 track DTR will be provided. No tape submittal is required. The DTR will be provided by the library.

* * * * * * * * * * *

\- - - REVISION - - -

AUTOCODER CONVERSION TO SYSTEM/360 MODEL 20 BAL

Two pages of Documentation and the Machine Readable Material have been changes.

Ordering Procedure: Program Order Number 360E-12.2.001

The program material can be obtained on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

If not specified, a 9 track DTR will be provided. No tape submittal is required. The DTR will be provided by the library.

* * * * * * * * * * *

OS/360 FORMATING STAND-ALONE CORE DUMP. In the event of an OS/360 system abend error, this program will provide a core dump that is almost identical to the standard OS/360 abend dump, only from a stand-alone environment. In addition, it will dump any other non-OS core image omitting the OS/360 control block formatting. All of core is dumped with the exception of the first sixteen bytes. At the end of the dump, core is restored with the original information and is saved so that in the event of a hardware failure both customer's program information and hardware logout information may be processed by this dump program and SEREP in that order.

Any messages to the operator are printed on the console typewriter and are self-explanatory. Machine check interrupts are masked off, and unlimited error recovery is incorporated so as to insure dump output if at all possible. Printed characters are limited to standard alphabetics and numerics to substantially speed up output to a UCS printer. Likewise, output may be selected to tape or printer on any System/360 or 1400 system. Operating System Required - OS/360. Minimum System Requirements - the only restrictions are that the program must be run from tape, requires any System/360 model 30 or larger, and must have a minimum core size of 32,768 bytes.

Ordering Procedure: Program Order Number 360D-04.1.009.

The program material can be obtained on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

If not specified, a 9 track DTR will be provided. No tape submittal is required. The DTR will be provided by the library.

* * * * * * * * * * *

A GANTT REPORT FOR MASTER PLANNING. This program produces a Gantt chart for master planning. Such charts are a very useful aid to effective scheduling and control of various aspects of production. Important events in the production cycle (such as critical deadline dates) are set off graphically on a time scale. The report produced by this program allows the user to look 15 weeks ahead of the current date. Options are provided to shift the time scale 1, 2, 3, or 4 weeks to the left. Thus, the report may be updated frequently with a minimum of keypunching new data. Programming Language - PL/I. Operating System Required - S/360 Operating System (OS). Minimum System Requirements - System/360, Model 30 or larger without modification. It is possible to modify the program so that it will run on smaller models of the System/360.

Ordering Procedure: Program Order Number 360D-15.4.005.

To obtain program material, submit one 2400 foot reel of magnetic tape. Specify whether 7 or 9 track recording is required. If not specified, 9 track recording will be used.

The required tape may be ordered from IBM or supplied with your request for the program.

* * * * * * * * * * *

A DOS GRAPHIC PROGRAMMING SERVICES PACKAGE FOR THE 2250. A graphic programming services package that supports the IBM 2250 Display Unit Model 1 in a DOS environment, provides nearly all the facilities of OS/360 Graphic Programming Services for the IBM 2250 Display Unit (GPS/OS). These facilities enable the ALP programmer to code programs for the 2250 using read-write logic, without concern for CCW's and data bit configurations. The apparent differences from GPS/OS are synchronous entry only to user attention routines, the limitation to one I/O operation at a time, and first-in-first-out dequeueing of attentions. Programming Language - All modules and routines are written in Assembly Language, and all macros are coded in the DOS Macro Definition Language. Operating System Required - DOS/360. Minimum System Requirements - The DOS minimum, the decimal feature, and a 2250 Display Unit, Model 1, with buffer. A DOS supervisor with TP-BTAM and the 2250 specified as a 2260 is required. No changes are made to any Type I programs. The logic module requires 4K bytes, the DTFGR requires 22 words, and other control blocks and macro expansions require storage dependent on the application.

Ordering Procedure: Program Order Number 360D-03.7.023.

The basic material can be obtained in card form or one one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

If not specified, a 9 track DTR will be provided. The DTR will be provided by the library.

The optional program material (Graphics Load Module) can be obtained on one 9-track or one 7-track (Data Conversion Feature Required) DTR. If the track requirement is not indicated 9-track will be forwarded. The DTR will be supplied by PID - no tape submittal is required.

* * * * * * * * * * *

IBM SYSTEM/360 SHORT CIRCUIT ANALYSIS PROGRAM FOR THE ELECTRIC UTILITY INDUSTRY. The Short Circuit Analysis Program simulates an electrical power system for power flows and voltages when abnormal operating conditions occur during a balanced or unbalanced fault on the system. The program uses the z matrix solution technique to simulate a maximum electrical transmission system consisting of 500 buses with 1000 transmission lines. Up to 50 buses may be retained by the program at one time.

Two types of fault conditions may be calculated by the program 1) a three phase fault study which may include either reactances only, or both resistances and reactances of the electrical system, and

2) a line to ground fault study (x only) which may include the mutual couplings within the electrical system. The information obtained from this program can be used to determine circiut breaker interrupting ratings as well as relay settings for the transmission system under consideration. Programming Language - FORTRAN IV. Operating System Required - DOS/360. Minimum System Requirements - S/360, 65K core storage, a card reader and a printer.

Ordering Procedure: Program Order Number 360D-16.4.008.

The program material can be obtained on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

If not specified, a 9 track DTR will be provided. No tape submittal is required. The DTR will be provided by the library.

* * * * * * * * * * *

CHECKPOINT/RESTART/RECREATE FOR S/360 OPERATING SYSTEM. Checkpoint/Restart/Recreate is a set of Assembly Language modules which runs under Operating System, MFT II, and has been integrated with releases 15.2 and 16. Primarily a tape-oriented checkpoint, it takes a checkpoint after the standard header labels of each reel of the data sets being checkpointed. The problem program may be restarted at any time from any of these checkpoints. At restart, the operator may re-assign devices. The restart program then issues mount messages, repositions all tape files, restores core and job queue information, and returns to the problem program. The recreate facility allows from 1 to 8 reels of a file to be reproduced, leaving the rest of that file and all other files intact. Checkpoints, restarts and recreates are essentially transparent to the problem program. Programming Language - Assembler Language. Operating System Required - OS/360 Release 15 and up. Minimum System Requirements - The program executes primarily out of the SVC transient area and requires not more than 1000 bytes of problem program core. It may be run on any hardware configuration under OS/MFT II and assembled under any OS System.

Ordering Procedure: Program Order Number 360D-03.0.006.

To obtain program material, submit one 2400 foot reel of magnetic tape. Specify whether 7 or 9 track recording is required. If not specified, 9 track recording will be used.

The required tape may be ordered from IBM or supplied with your request for the program.

* * * * * * * * * * *

DOUBLE 7074 EMULATION ON 360-65-I. This is a 360 ALP program which allows the S/360-65I with shared storage RPQ and 7074 emulation feature to con-currently operate two 7074 programs in emulation mode. This program is not model dependent i.e. 360-65. If such an RPQ were available on a Model 50, the program would also be able to run.

The two jobs running are logically independent and unaware of each others existence. They operate in a time-slicing mode based upon I/O interrupts. Each program has dedicated channels corresponding to the 7074 programs. Two consoles are supported. The primary resource shared is CPU time. 360-65I users who have 7074 emulation and 360-65H users who need to increase core for growth should review applicability of this program and the EMOS (7074/OS) program. They give emulator users the ability to multiprogram 7074 runs either with other 7074 runs or with OS without reprogramming gaining the advantage of overlapping setup, typing, rewinding and CPU as available with other productive computing. Programming Language - 360 ALP, and requires IBM Program 360C-EU-725 Version 3. Minimum System Requirements - Program requires one tape for loading or program which is subsequently available to the system. The user will probably double his 7074 I/O configuration having for example, two typewriters, two card readers, 4 channels with 4 or 5 tape drives on each channel. No disk is used. The exact configuration will vary with the hardware requirements of the 7074 programs to be run.

Ordering Procedure: Program Order Number 360D-05.2.003

The program material can be obtained on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

If not specified, a 9 track DTR will be provided. No tape submittal is required. The DTR will be provided by the library.

* * * * * * * * * * *

FORTRAN STATEMENT NUMBER SEQUENCING FOR 1130, 1800. Process a FORTRAN Source Language Program, sequencing the statement number labels and all references to the labels in the program, according to input specifications. Optional punch, list. Cross-reference of old to new statement numbers. Also uses new ID specification. Programming Lanugage - FORTRAN. Minimum System Requirements - Uses 7,610 words of core. Also, the program requires 126 disk blocks excluding the IDEAL and Commercial routines. I/O devices required - 1442 Card Reader/Punch and the 1130 Printer.

Ordering Procedure: Program Order Number 1130-03.6.001.

Distribution will be in card form.

* * * * * * * * * * *

OPTIMIZATION OF MATERIAL FOR MAXIMUM CABLE PRODUCTION. The program provides a machine loading schedule for the production of multi-conductor cables where an individual conductor length choice is needed. While meeting various order specifications the program provides maximum cable length production and if the order cannot be filled with available wire, it gives requirements for complete order. At the same time it gives minimum waste, mimimum set-ups, minimum spool changes and a maximum of long cable lengths. The program also provides information for cost analysis and a better inventory system for wire available. Inputs to the program include spool lengths according to conductor type, cable lengths desired, mimimum acceptable cable lengths, total ordered footage, waste point, and twist loss. Programming Language - Source - Basic FORTRAN IV. Monitor System Required - 1130 Disk Monitor, Version 2, Mod.1. Minimum System Requirements - 1130 with 8K core, card input and printer output.

Ordering Procedure: Program Order Number 1130-23.1.005.

Distribution will be in card form.

* * * * * * * * * * *

ROMANCE - PRATT AND WHITNEY TAPE-O-MATIC POST PROCESSOR. The ROMANCE/TAPE-O-MATIC POST PROCESSOR program is designed to use the output of ROMANCE (1130-23.4.002) to produce punched paper tape for the Pratt and Whitney Tape-O-Matic Numeric Drilling Machine, Model B. The tape format is tab sequential and a listing of the tape is provided. Console switch options provide card output in lieu of paper tape, debug facility and printing by pass. A utility program is included with the package to reread and list the tape. Programming Language - FORTRAN IV. Operating System Required - 1130-OS-002, V1M2. Minimum System Requirements - 1131 Model 2B (8K-Disk) 1442 Card/Read Punch.

Ordering Procedure: Program Order Number 1130-23.4.005.

Distribution will be in card form.

* * * * * * * * * * *

MODIFICATIONS TO 1130 TCP TO DRIVE LINOFILM QUICK. The 1130 Type Composition Program (DP-04X) is used extensively to drive hot metal linecasting machines. Many printers who have replaced this equipment with the Linofilm Quick device can use the 1130 Type Composition package plus the minor changes outlined in this program to provide continued use of computerized typesetting in their shop. These patches to the 1130 Type Composition Program, Version 1, Level 2 enable the user to drive the Linofilm Quick device in a manner similar to the techniques used in driving a 4-magazine linecaster. Programming Language - Source - 1130 Assembler Language. Minimum System Requirements - Those required to run the 1130 Type Composition Program (DP-04X).

Ordering Procedure: Program Order Number 1130-29.4.001

Distribution will be in card form.

* * * * * * * * * * *

1130 FUNCTION PLOT ON PRINTER. This is a demonstration program primarily for educational institutions, but can also be used by them in a math or physics curriculum. It demonstrates the high internal speed of the 1130, the use of IBM-supplied subroutines such a Sine, Log, etc., the advantages of a disk machine, and the flexibility of man-machine interaction. The user inserts his function card and data card/s/. The program plots the function on the printer, then asks for changes. It also handles undefined regions of a function. Programming Language - FORTRAN with one Assembler subroutine. Monitor System Required - 1130 Monitor. Minimum System Requirements - 8K disk 1130 with 1442 and 1132.

Ordering Procedure: Program Order Number 1130-30.2.002

Distribution will be in card form.

* * * * * * * * * * *

1800 MACRO UPDATE PROGRAM. This program is a Macro Update Program (MUP). It is to be used in conjunction with the 1800 Macro Assembly Program (File Number 1800-03.7.001). This program permits updating of system macros defined by the referenced processor. The program permits insertion and deletion of statements within a macro or of entire macros with a single statement. It also permits the removing and renaming of macros. The macro update program facilitates the maintenance of large or small programs, residing on disk and system macros. Programming Language - 1800 Assembler. Monitor System Required - TSX II or TSX III. Minimum System Requirements - 4200 words of variable core; this is the only configuration restriction.

Ordering Procedure: Program Order Number 1800-03.6.001.

Distribution will be in card form.

* * * * * * * * * * *

NEW TYPE IV PROGRAMS

The following are the abstracts of programs which have been recently made available from the Type IV library.

The program along with its complete abstract, will be incorporated in subsequent issues of the Catalog of Programs.

Programs may be obtained by submitting a properly completed "General Program Request Card" (Form Number 120-1145) to the Program Information Department, 40 Saw Mill River Road, Hawthorne, New York, 10532.

These programs and their related documentation are distributed by IBM in the author's original form and have not been subjected to any formal testing.

Any discussions of Type IV programs with customers must emphasize the following points:

1. - Type IV (IBM customer-contributed) programs are provided by the IBM Corporation as part of its service to customers.

2. - Type IV programs have not been subjected to any formal test.

3. - Recipients of Type IV programs are expected to make the final evaluation as to the usefulness of the programs in their own environment.

4. - There is no committed maintenance for Type IV programs. However, any changes the author chooses to make will be announced in subsequent issues of the Catalog of Programs.

5. - IBM makes no warranty, expressed or implied, as to the documentation, function, or performance of Type IV programs.

* * * * * * * * * * *

\- - - REVISION - - -

COOLEY - TUKEY FAST FOURIER TRANSFORM

All pages of the Documentation and the Machine Readable Material have been changed.

Ordering Procedure: Program Order Number 360D-13.4.002

Distribution will be in card form.

* * * * * * * * * * *

COOLEY-TUKEY FAST FOURIER TRANSFORM ON DIRECT ACCESS. Subroutine FOR2D (IDATA, N, NDIM, ISIGN, IFORM, WORK, NELEM) for N data points, FOR2D runs in time proportional to N log N, while pre-FFT methods run in N**2. For large N, this is seconds versus hours. The FFT is the fastest and most accurate way of computing periodograms of time series, convolutions and correlations, Fourier coefficients, digital filters, partial differential equations, etc. DATA is a complex, multi-dimensional array of data kept on direct access storage in file IDATA, in records of length NELEM. N is an array of length NDIM giving the dimensions of DATA. NDIM is unrestricted; each N(I) must be a power of two, as must NELEM. ISIGN = +1 or -1, the transform direction. IFORM = 1. WORK is a one-dimensional complex array of length 3*NELEM used for input/output buffers. The user must supply two subroutines, DREAD and DWRIT, for reading and writing records in arbitrary order to and from the direct access storage device (disk, drum, etc., but not tape). Programming Language - USASI Basic FORTRAN. Operating System Required - BOS, DOS, or OS. Minimum System Requirements - S/360, Model 30 with 64K core storage, card reader, disk files, and printer.

Ordering Procedure: Program Order Number 360D-13.4.006

The program material can be obtained in card form or on one 9 track or 7 track (Data Conversion feature required) Distribution Tape Reel (DTR).

If not specified, a 9 track DTR will be provided. No tape submittal is required. The DTR will be provided by the library.

IBM Installation Newsletter

April 25, 1969

Issue No. 69-08

CONTENTS

(G means Green Section)

Distribution: Branch Office - DP Management, Salesmen, Systems Engineers, FE Managers. Regions, Districts, Education Centers, Field Systems Centers, Federal Systems Centers, FE Area Offices, DPD HQ, FED HQ, WTC

*Requires Immediate Attention

SELECTED SHORT SUBJECTS

The purpose of Selected Short Subjects is to bring together and highlight concise, factual and timely information which will indicate that action is to be taken by the IBM representative whose accounts are affected.

1. New Newsletter Feature

A new feature to help you find New Type III and IV Programs has been added to the INL. Each issue will contain an updated Accumulative List of New Type III and IV Programs. This should help cut down your search time. Take a look at the end of the green pages for further details.

2. Suggestions for Completing the New PID Program Order Form

Judging from the first arrivals of the new Program Order Form at PID, several of the instructions have not been made clear. The most frequent problems arise in the following areas of the form:

Section 1, Columns 1 and 2

These columns must be filled in for each program ordered so that we know whether the requester wishes the Basic or Optional material and the Machine Readable material or only the Documentation.

Section 1, Column 3

Documentation only orders for Type I and II Programs should be sent to Mechanicsburg, not PID.

Section 1, Column 14

Program Update Service is provided automatically to users ordering Type I and II Programs. Through the Program Update Service, users of record get automatic distribution of program maintenance. Users of record also get announcements of total replacements and/or maintenance packages which must be ordered from PID.

An action code "M" in column 14 denotes that a user of record is ordering an announced maintenance package for a program or a programming system. An action code "R" in Column 14 denotes that a user of record is ordering an announced total replacement for a programming system. These are exceptions to the normal ordering pattern. Ordinarily, Column 14 is left blank.

In any case, do not use Column 14 to request Program Update Service.

Section 1, Columns 21-22

The medium on which each program is available from PID is identified in the Catalog of Programs. The code for the desired medium should be entered in Columns 21/22. This field should not be left blank if Machine Readable material is ordered.

Section 1, Columns 23-24

When a program is available from PID on a User Volume (Tape/Disk), the number of User Volumes required should be entered in Columns 23-24 and the total number required to fill the order should be accounted for in Section 3. This field should be left blank if User Volumes are not required.

Section 3

If User Volumes are required, as determined in Section 1 Columns 23-24, the total number of tapes and/or disks and the manner in which they are being provided to PID should be entered. If no User Volumes are required, Section 3 should be left blank.

Section 4

No entry at all is required in Section 4 for a normal order for a user who has ordered programs from PID before. But if any data is entered in Section 4, an action code must appear.

To register a new user, enter action code "R", the Branch Office or the IBM Division/Location/Department (not both), the user's S/360 Magnetic Tape Capability and the "Ship To" information. To modify an existing registration, enter action code "M" and the new information in the block to be modified.

The action code "S" should be used only when the "Ship To" information for this particular order is different than the information on the PID user record. Use "S" only if the User is already registered with PID. The information which should be entered in Section 4 when action code "S" is used is "Ship To" information.

Section 5

An IBM Management Signature, the date and the Branch Office/Department Number should be entered in Section 5. The Special Messages area should be used only for Liquidated Damages information.

Complete instructions on the new Program Order Form appear in all Catalogs of Programs, in ~~the~~ the Branch Manual (Sales Activity section), on the cover of all Order Form pads, and on the back of each Order Form set. These instructions should be reviewed and understood before submitting program orders to PID.

OS/360 RELEASE 17 MVT START COMMAND VALID PROGRAM NAMES

MVT users who have added program names to the Start Command Program list (module IEEVLNKT) should be aware that Release 17 has added four new module names to the list. The complete list in module IEEVLNKT is as follows:

IEFIRC	Reader
IEFSD080	Writer
IEEVMNT2	Mount
IHKRJBGN	RJE
IEFVRRC	CHKPT/RESTART
IKDINPPO	Graphics (SGJP)
IKDSGJP	Graphics (SGJP)
IEF VMA	ASB Reader
IKAGFX	Graphics (GJP)
IKAGJP	Graphics (GJP)
IEFSD060	INIT

Any program that is found in the EXEC statement of a Start procedure must appear in this list for the procedure to be successfully started.

OS/360 MVT HIERARCHY STORAGE SUPPORT

Main Storage Hierarchy support is the programming support for the IBM 2361 Core Storage. It provides selective access to either processor storage or the 2361.

Main Storage Hierarchy Support divides main storage into two blocks, known as hierarchies. Hierarchy 0 is assigned to processor (high speed) storage and hierarchy 1 to the 2361.

If 2361 Core Storage is not attached to the computing system the hierarchy structure will be maintained by allocating hierarchy one storage requests in processor storage. (For a detailed description of this support you should refer to IBM SRL C27-6942 entitled "Introduction to Main Storage Hierarchy Support for IBM 2361 Models 1 and 2".)

The following keyword parameter, added to the CTRLPROG macro instruction at system generation, makes the storage hierarchy feature a part of the system:

HIARCHY =(Include)
(Exclude)

Job Control Language parameters allow the user to specify the size of the region segments (hierarchy 0 and 1) to be obtained for execution of each job step.

, REGION=(value 0K , value 1K)

Where value 0K refers to the size of the region segment in hierarchy 0 and value 1K refers to the size of the region segment in hierarchy 1. e.g. REGION =(80K, 120K) indicates that 80K of hierarchy zero storage and 120K of hierarchy one storage be allocated for this region request.

It is important to note the following operation considerations related to an MVT environment.

1. The hierarchy structure will be maintinaed even though the 2361 is not attached to the computing system. For example, a REGION=(80K, 120K) request will result in two region segments (of 80K and 120K) being allocated from processor storage.

2. If the REGION parameter is omitted from the Job Control Language, a default value will be supplied by the Reader/Interpreter.

   This default value will be from hierarchy zero. There is no means of specifying a hierarchy one value in the Reader/Interpreter default options.

3. Regardless of the hierarchy zero size requested through Job Control Language, a hierarchy zero segment large enough to accommodate the Initator will be obtained to initiate the job step. If the terminator is resident in the system Link Pack Area the "initiator size region" may be exchanged for a smaller one (minimum size of approximately 12K) at the time the processing program is given control. This is done so that the job scheduler is guaranteed to have high speed storage in which to operate, so that potential overrun situations may be avoided.

4. System tasks, such as Reader/Interpreters, Output Writers and Remote Job Entry cannot be "STARTED" into a hierarchy one region segment. In general, any function which is initiated with a START command cannot be directed to hierarchy one.

5. Modules from the system linkage library (SYS1. LINKLIB) or from the system SVC library (SYS1. SVCLIB) may be relinked with the hierarchy one attribute and may then be loaded, by NIP, into the appropriate hierarchy of the Link Pack Area. In this manner it is possible to cause system functions to reside in LCS, however, all GETMAIN requests issued by these modules must be satisfied from hierarchy zero.

6. If it is required to be able to execute a processing program from either hierarchy zero or hierarchy one it is necessary to have two copies of the load modules in the program library. One copy having the hierarchy zero attribute and the other one having the hierarchy one attribute. This will tend to increase the size of program libraries and of procedure libraries, since different program names must be used.

OS/360 RELEASE 18 MCS CHANGES PLANNING INFORMATION

For Release 18 Multiple Console Support, a SYSGEN option, SVC 34 was changed so that it would expect a value in Register 0 indicating the console source of the command to be processed. This value is used to determine whether or not the console which issued the command has the authority to issue that command.

Register 0, therefore, should contain a zero in order to process internal commands. An internal command is defined as being a command which was not issued by the operator nor was introduced into the system via the input stream. A zero value in Register 0 indicates that the invoker of SVC 34 can issue all commands. If a value other than zero is in Register 0, the command would be executed in a non-MCS environment, but might be ignored in an MCS environment and an error message may be issued by SVC 34.

DOS/360 NEW ENDICOTT GUIDELINES

Endicott Guidelines 'R' is available as TIE paper Z77-8084-1. This covers the planning and implementation of a DOS system.

Endicott Guidelines 'R-1' is an update to 'R' and covers information to generate DOS Release 19. The procedure outlined in Guidelines 'R-1' is applicable to generation of Release 20. Guidelines 'R-1' is available as TIE Z77-9059.

Endicott Guidelines 'T' explains multitasking under Release 19. Included are four working program examples demonstrating the use of multitasking. Guidelines 'T' is available as TIE Z77-9006.

Editor's Note: Remember that TIE papers are "For IBM Internal Use Only".

INFORMATION ABOUT THE NEWSLETTER

EVERY SALESMAN AND SYSTEMS ENGINEER SHOULD RECEIVE ONE COPY OF THE NEWSLETTER. Each FE Branch Manager should receive five copies for distribution to Customer Engineers.

The IBM Installation Newsletter is distributed only to IBM locations and is not available to customers. The Newsletter is separated into two parts. The first part contains information which is intended only for IBM personnel and the pages are labeled For IBM Internal Use Only. THESE PAGES, OR REPRODUCTIONS OF THEM, ARE NOT TO BE GIVEN TO CUSTOMERS. However, using your own good judgement, you may discuss pertinent information from these pages with a customer. The second part contains information which may be REPRODUCED by the Branch Office at their discretion and given to customers. These pages are not labeled with any restrictive classification. Note that when any abstracts for Type III programs are given to the customer, he must be informed that the disclaimer for Type III programs as contained in the Newsletter, applies. YOUR COOPERATION IN PROPERLY HANDLING THE CONTENT OF THE IBM INSTALLATION NEWSLETTER IS NECESSARY FOR ITS CONTINUED PUBLICATION.

Changes to the quantity of the Newsletter shipped are to be made by the Administration Manager. (NOTE THAT EXTRA COPIES OF THE NEWSLETTER ARE NOT TO BE ORDERED FOR DISTRIBUTION OF INFORMATION TO CUSTOMERS.) He should update and send the DP Administration Managers' Document Control card (Electro IBM N61508) to DP Document Control, DPD HQ.

Direct inquiries concerning the contents of this Newsletter to:

W. F. Malmborg, Editor
IBM Installation Newsletter
Dept. 812, DPD HQ
112 East Post Road
White Plains, New York 10601
Telephone 914 Wh9-1900 Extension 6158
Tie Line 374-6158

NEW PUBLICATIONS ANNOUNCED BY PRL'S

Information is extracted and condensed from the weekly PRL's (Publications Release Letters) to insure that all Salesmen and Systems Engineers are aware of new or revised Reference Sources, Marketing Publications, and Field Engineering Publications. Normally, each issue of the Newsletter will contain information extracted and condensed from two PRL's, one following the other. The information will be placed in the Newsletter in its original sequence with no rearrangement of form numbers or titles. It is not intended to replace existing information and distribution sources. You should be certain that you are aware of these sources.

Do not use the condensed Newsletter information to order publications. Most but not all (for example, some films and slide sets) of the items are available from Mechanicsburg. To receive the items you need without delay, prior to ordering check the REQ (Requisition Form) code in the "Accumulative Supplement to IBM Publications Current Price List (Z20-0100)" which is attached to the weekly Publications Release Letter (PRL) and distributed to your Administration Manager. For an explanation of the REQ codes and other publications information see the first few pages of the "IBM Publications Current Price List Z20-0100.

The Publications Requisition, form number M02-0618 is used for ordering items from the IBM Distribution Center, Mechanicsburg, Pa.

(IBM World Trade Corporation has its own distribution center and procedures.)

PRL #15 April 11, 1969

REFERENCE SOURCES

FORM NO.	TITLE	DISPOSITION
N20-0360,56	IBM S/360 Newsletter Re: A22-6822-12	Scrap N20-0360,55
N20-1869-1	TNL to BAP Insurance Re: H20-0536	Scrap N20-1869-0
Z20-1930-0	KWIC Tie March 1969 Supplement Re: Z20-0305-20	NEW

MARKETING PUBLICATIONS

221-0365-2	The IBM 1442 N1 Card Read Punch	NEW
221-0372-1	The IBM 2321 Data Cell Drive	NEW
221-0405-2	The IBM 2560 Multi-Function Card Machine	NEW
221-0421-2	2401 Magnetic Tape Unit, Model 4, 5,6	Scrap 221-0421-1
221-0487-2	IBM 2495 Tape Cartridge Reader	Scrap 221-0487-1
320-1906-3	March 1969 In Brief	Scrap 320-1906-2
322-0045-0	IBM Journal of Research and Development, Vol. 13, No. 2	NEW
520-1455-1	1130 Computing System for the Construction Industry	Reinstated Abstract
520-1532-2	Teleprocessing Support Services	Scrap 520-1532-1
520-1579-0	1287/1288 OCR	NEW
520-2146-0	Maintenance Control for Airlines	NEW
520-2159-0	Secom - Systems Engineering Communication	NEW
549-0132-0	Baskerville Type Style Family	NEW
549-5003-0	Graphics Literature for Sale Kit	NEW
A32-0007-2	2420 Model 5 Magnetic Tape Unit.	Scrap A32-0007-1
H20-0513-1	Linear Programming System/360 (LPS/360) (360A-CO-18X) Application Description	Scrap H20-0513-0 & N20-1877
H20-0607-0	Linear Programming System/360 (LPS/360) (360A-CO-18X) Program Description Manual	NEW
H20-0611-0	Linear Programming System/360 (LPS/360) (360A-CO-18X) Operations Manual	NEW
K20-0274-0	IBM 1800 Plant Automation System for Real Time Production Monitoring at Sylvania Lighting Products Division	NEW - Reinstated
K20-0279-0	Holiday Inns of America, Inc. -- HOLIDEX Reservation System Using the IBM S/360	NEW
K20-0309-0	Punched Card Data Processing at the M. P. Heinze Company	Abstract only
L22-6904-0	IBM S/360 Mod 40 RPQ Table Address Chaining	NEW
N23-0683	Changes to S/360 Basic PL/I Coding Workshop Instructor Guide Re: R29-0118-2	NEW
N24-0427	IBM S/360 Model 25 Operating Procedures Re: A24-3523-0	NEW
N27-3035	Changes to "Planning and Installation of a Data Communications System Using IBM Line Adapters" Re: A24-3435-2	NEW
N28-0251	IBM S/360: FORTRAN IV Language Re: C28-6515-7	NEW
N33-9051	TNL to Model 20 CPS 1419 IOCS Re: C26-3607-1	NEW
R20-4110-0	IBM S/360 Matrix Language (MATLAN) Education Guide	NEW
R20-9240-0	IBM S/360 Remote Access Computing System (RAX) Course Description	NEW
R20-9247-0	IBM S/360 DOS Workshop Course Description	NEW
R20-9276-0	IBM S/360 Project Management System (PMS) Course Description	NEW
R20-9291-0	IBM S/360 Mathematical Programming System Report Generator Course Description	NEW
X21-9093-0	Calculation Specifications	NEW - Scrap X24-3351
X21-9094-0	Input Specifications	NEW - Scrap X24-3350
X74-6259-1	Layout Form	Reinstated
Y20-0346-0	IBM S/360 Problem Language Analyzer (PLAN) (DOS/OS) (360A-CX-26X, 360A-CX-27X) Vol. II Flowcharts System Manual	NEW
Y23-9002-0	Teleprocessing Analysis and Design Program (TPAD) Course Description	NEW
Y26-1581-0	IBM 1500 Physical Planning Template	NEW
Z20-1905-3	Inserts (less programming)	Scrap Z20-1905-2
Z20-1906-3	Inserts DP Sales Manual Programming Section	Scrap Z20-1906-2
Z20-1907-1	Managers Manual Binder Index Tabs	Use Z20-1907-0
Z77-9004-1	Coding Basic PL/I Efficiently	Scrap Z77-9004-0
Z77-9106-0	An Operating Guide for Power, A Type III DOS Spooling Program	NEW
Z77-9107-0	Use of the Basic Counter Unit to Measure Channel Usage	NEW
Z77-9108-0	A Comparison of Features of PCS, PMS & Project	NEW
Z77-9109-0	Techniques for Programming the 6405 Card Readers	NEW
Z77-9110-0	DOS BOMP Implementation Planning, an Illustrated Branch Office Lecture for Salesmen and Systems Engineers	NEW

FIELD ENGINEERING PUBLICATIONS

123-0461-1	IBM 2314 IPC	Scrap 123-0461-0
123-1029-0	2845 Display Control (Preliminary) Parts Catalog	NEW

PRL #15 (Continued)

131-0548-3	2780 Data Transmission Terminal IPC	Scrap 131-0548-2
131-0565-0	1907-6 Batch Recording System IPC	NEW
229-0015-0	Domestic Temporary Assignment Administrative Guide for Employees	NEW
229-1287-1	Think Safety Label	Use 229-1287-0
229-4039-1	IBM S/360 Ready Reference	Use 229-4039-0
229-7314-18	DP Maintenance Parts Price List	Scrap 229-7314-17
229-8003-0	IBM 360 Model 20 Storage Control FE Reference Card	NEW
R23-3029-3	2321 Data Cell Drive Student Guide	Use R23-3029-2
R23-3062-3	360 Introductory Program Student Self Study Course	Use R23-3062-2
R23-3131-0	2415 Magnetic Tape Unit & Control Student Self Study Course	NEW - Use Y25-2155-0 & Y25-2168
R23-3156-0	2915 Display Terminal Mod 3 Student Self Study Course	NEW
R23-3157-0	2841 Storage Control Unit Student Self Study Course	NEW
R23-3158-0	2065 P U External Diagram Student Self Study Course	NEW
R25-5364-1	2703 Transmittal Control B Student Guide	Scrap R25-5364-0
R31-0324-0	Class Enrollment Sheet	NEW - Reinstated
V31-0113-0	2060 External Diagnostics Microfiche Card	NEW
V31-0118-0	2025 FIS System I/O Microfiche Card	NEW
V31-0127-0	2841 Storage Control Microfiche Card	NEW
Y22-6601-2	2365 Processor Storage FE MDM	Use Y22-6601-1
Y24-0096-0	FES to 2025 FEMM Re: Y24-3528-0	NEW
Y24-0099	2025 Processing Unit Integrated 1403 Attachment Model N1 Supplement Re: Y24-3533-0	NEW
Y24-0100	Supplement to IBM S/360 Model 25 FEMM Re: Y24-3528-0 with Y24-0096	NEW
Y25-2176-1	2970 Reservation Terminal Administrator Guide	Use Y25-2176-0
Y25-2203-0	360 Prog. TOS/DOS Prerequisite Quiz (Student Quiz)	NEW
Y25-2700-3	360 Model 40 System I/O Part I Administrator Guide	Scrap Y25-2700-2
Y25-2750-0	Supplement to Y25-2719-0 Student Guide	NEW
Y25-2756-1	2311 Disk Storage Drive Model 20 Student Quiz #1	NEW
Y25-2757-1	2311 Disk Storage Drive Model 20 Student Quiz #2	NEW
Y25-2768-0	2841 Storage Control Unit Student Quiz #1	NEW
Y25-2769-0	2841 Storage Control Unit Student Quiz #2	NEW
Y25-2770-0	2841 Storage Control Unit Student Quiz #3	NEW
Y25-2771-0	2841 Storage Control Unit Student Quiz #4	NEW
Y25-2772-0	2841 Storage Control Unit Student Quiz #5	NEW
Y25-3117-0	2313 Comprehensive Quiz (Student)	NEW
Y25-3118-0	2311 Comprehensive Quiz (Student)	NEW
Y25-3816-1	1907 Batch Rec System Model 5 Administrator Guide	Scrap Y25-3816-0
Y25-3817-1	1907 Batch Rec System Model 5 Student Quiz	Scrap Y25-3817-0
Y26-1576-0	IBM 1130 FETM (Telpak RPQ)	NEW
Y26-3617-S	IBM 1800 FETO	Use Y26-3617-4 & Y26-0620
Y27-0008-3	2703 Transmission Control FEMDM	Scrap Y27-0008-2
Y27-2143-3	2915 Display Terminal Model 3/2948 Display Control Model 4 FEIM	Scrap Y27-2143-2
Y27-2236-0	2989 Basic Counter Unit FETMIM	NEW
Y33-1034-0	2560 Multi-Function Card Machine Attachment Feature, 2501 Card Reader Attachment Feature, 1442 Card Punch Attachment Feature System/360 Model 20 (Machines with serial no. 50,000 & above) FETOM	NEW
		NEW - Scrap Z33-1034-0

PRL #16 April 18, 1969

FORM NO.	TITLE	DISPOSITION
N20-7070.36	7070 SRL Newsletter Re: A22-6699-4	Scrap N20-7070.35

MARKETING PUBLICATIONS

221-0518-0	Design and Analysis of Communictions - based Systems	NEW
520-2155-0	Data Processing for Public Accounts	NEW
545-0096-0	Government Price List	NEW
549-0710-1	Guide Copy Prep Sample Projects	Scrap 549-0710-0
549-5001-0	Composer Type Style Brochure	NEW
549-5004-0	Magnetic Tape "Selectric" Composer Folder	NEW
549-5005-0	"Selectric" Composer Folder (Saleable)	NEW
570-0208-8	Term Contract for Tabulating Cards	Scrap 570-0208-7
A21-9031-2	IBM S/360 Component Description and Operating Procedures 1231-N1	Scrap A21-9031-1, N21-0061, N21-0068 & N21-0089
A24-3411-2	IBM S/360 Model 30 Channel Characteristics and Functional Evaluation	Scrap A24-3411-1 & N24-0349 & A24-3411-0
A32-0004-0	2420 Model 7 Magnetic Tape Unit	NEW
C21-5002-1	1130 RPG Specifications	Scrap C21-5002-0
C24-5034-2	IBM S/360 TOS System Control & System Service Programs	Use C24-5034-1, N24-5219, N24-5290 & N24-5348
N20-1945	TNL for Information Management System/360 for the System/360 (System Description) Re: H20-0524-0 and -1	NEW

PRL #16 (Continued)

N20-1958	TNL for System/360 AD-APT/AUTOSPOT (OS) Numerical Control Processor (360A-CN-12X) Part Programming Manual Re: H20-0549-0	NEW
N21-5102	IBM S/360 Disk and Tape Operating Systems Report Program Generator Specs. Re: C26-3570-4, -5	NEW
R20-4125-0	IBM S/360 Remote Access Computing System (RAX) Education Guide	NEW
R20-9242-2	IBM S/360 Basic PL/I Coding Workshop	Scrap R20-9242-1
R29-0019-1	IBM Fundamentals of Programming Text P.I.	Use R29-0019-0
R29-0117-2	IBM S/360 PL/I Coding How to Write a Basic PL/I Program Examination	Use R29-0117-1
R29-0266-0	IBM S/360 PL/I (F) Coding P.I Course	NEW
V25-6466-0 thru V25-6490-0	7070-7074 Data Processing System - Programming Video Tape Course	NEW
V25-6705-0	Fundamentals of Programming Workshop (Video Tape)	NEW
V25-8200-0	List Directed Transmission in Stream-Oriented Input/Output for Basic PL/I (Video Tape)	NEW
X20-1704-1	2321 Reference Card	NEW
X24-3541-0	IBM S/360 Model 25 Channel O Worksheet	NEW
Y20-0044-1	Index Tabs (6)	Use Y20-0044-0 & N20-1897
Y20-0083-2	Project Management System/360 (360A-CP-04X) Version 2, Network Processor System Manual	Use Y20-0083-1, Y20-0218, Y20-0083-0, Y20-0166
Y20-0363-0	1130 Program for Optical System Design/II (Posd/11) (1130-EO-14X) System Manual - Assembly Listings	NEW
Y20-0364-0	System/360 Program for Optical System Design/II (POSD/II) (OS) (360A-EO-15X) System Manual Assembly Listings	NEW
Y20-0365-0	IBM S/360 Program for Optical System Design/II (POSD/II) (DOS) (360A-EO-16X) System Manual Assembly Listings	NEW
Y20-0370	TNL for S/360 APT Numerical Control Processor (360A-CN-10X) Version 4 - System Manual - Flowcharts Re: Y20-0227-0	NEW
Y20-0377-0	Introduction to IBM S/360 Model 65 Multiprocessing	NEW
Y20-0380	TNL to Communications Network Design Program/360 CNDP/360 DOS/TOS (360A-SE-28R) Program Description & Operations Manual Re: Y20-0207-1	NEW
Y23-5900-0	Practice Sales Call Demonstration - Video Tape	NEW
Y23-8200-0	TV Applications in DP - Video Tape	NEW
Y23-8201-0	DOS Update Release 19 (Reel No. 1)	NEW
Y23-8202-0	DOS Update Release 19 (Reel No. 2)	NEW
Y23-8203-0	DOS Update Release 19 (Reel No. 3)	NEW
Y23-8204-0	Model 85 Performance Measurements January 1969 (Video Tape)	NEW
Y23-8205-0	OS/360 Accounting Review (Video Tape)	NEW
Y23-8206-0	OS/360 System Management Facilities (Video Tape)	NEW
Y23-8207-0	OS/360 PL/I (F) Object Program Management (Video Tape) Reel No. 1	NEW
Y23-8208-0	OS/360 PL/I (F) Object Program Management (Video Tape) Reel No. 2	NEW
Y33-8600-0	PL/I Compile-Time Facilities (Video Tape) Reel No. 1	NEW
Y23-8601-0	PL/I Compile-Time Facilities (Video Tape) Reel No. 2	NEW
Z20-1929-0	Information Management System/360 Marketing Guide	NEW
Z20-1959-2	Field Directory Revision	Scrap Z20-1959-1

FIELD ENGINEERING PUBLICATIONS

123-0420-3	2050 Processing Unit Parts Catalog	Scrap 123-0420-2
229-0014-1	FE Microfiche Handbook	Scrap 229-0014-0
229-5057-0	FE Diagnostic Programs Handbook	NEW
BOF-5317-0	2040/2050 External Diagnostics	NEW
R23-3092-4	System/360 Model 40 System I/O Parts I and II Student Self Study Course	Use R23-3092-3
S23-4064	FE Supplement to 29 Card Punch FE Theory of Operation Re: 225-3358-3 to -2, S23-4064	NEW
S31-0566	1287 Optical Reader Re: 131-0552-3 Parts Catalog	NEW
V31-0132-0	DOS Emulator for M 30/40 Microfiche Cards	NEW
Y22-6772-0	IBM S/360 Model 40 1401/1460 Compatibility Feature Re: Y22-2859-0 FETOM	NEW - Scrap Y22-6767
Y24-3535-0	2025 Processing Unit Integrated 2560 Attachment FETOM	NEW
Y25-2170-0	IBM S/360 Prog TOS/DOS Module I Student Quiz	NEW
Y25-2171-0	IBM S/360 Prog. TOS/DOS Module I Student Quiz	NEW
Y25-2172-0	IBM S/360 Prog. TOS/DOS Module I Student Quiz	NEW
Y25-2766-0	2915 Display Terminal Mod 3 Student Quiz	NEW
Y25-2767-0	2915 Display Terminal Mod 3 Administrator Guide	NEW
Y27-2145-2	2915 Display Terminal Model 3 (RPQ E 41062) 2948 Display Control Model 4 (RPQ 880629) FEDM	Scrap Y27-2145-1
Y27-2239-0	2915 Display Terminal/2948 Display Control FES to Y27-2146-1 FEMM	NEW
Y27-2240-0	IBM 2250 Display Unit Model 1 FES to Y27-2045-1 FEMM	NEW
Y33-1027-0	2203 Printer Attachment Feature S/360 Model 20 FETOM (M/c with serial no. 50,000 and above) FE Description of Operation	NEW - Scrap Z33-1027-0
Y33-1044	FES to FETOM 2520 Card Read Punch Attachment Feature, System/360 Model 20 (machines with serial no. 50,000 and above) Re: Y33-1029-0	NEW
Z22-2828-3	IBM S/360 Model 50 Main Storage, Local Storage, Storage Protection FETOM	Use Z22-2828-2 & Z22-6769

April 25, 1969 **IBM INSTALLATION NEWSLETTER** Issue No. 69-08

Table of Contents

OS/360 SYSGEN UNDER MVT

The intent of this article is to clarify the support for SYSGEN under MVT and to explain what the user must do. There are two basic ways of doing a SYSGEN under MVT. One way is to break the Stage II output into different jobs. The other way is to run Stage II as a single job. This article will address both of these approaches. Those interested in leaving the Stage II as a single job should read the section "General Considerations" and then the section on "Stage II as a single job". Those who are interested in breaking the Stage II into multiple jobs should also read the section on "General Considerations" as well as the section on "Stage II as multiple jobs" All users should carefully review the memo to users as well as the appropriate sections of the SRL "OS/360 System Generation", C28-6554-5.

General Considerations

In order to use your own system as a driver for SYSGEN you must make a new catalog entry in the driver system for each Component Library (such as SYS1.CI505) required by SYSGEN or the portion of a SYSGEN you are doing. The new entry will point to SYSGEN volumes and may change standard catalog entries in your system. If SYS1.COBLIB (for COBOL E) SYS1.FORTLIB, SYS1.PL1LIB or SYS1.SORTLIB are to be included in the new system then you will have to change your catalog entries for these data sets. On this topic, see the last paragraph on Page 17 of the SRL "OS/360 System Generation". You should also change your catalog for SYS1.MACLIB to point to the appropriate SYSGEN volume. Many of the data sets that exist in your driver system will not have to be recataloged such as SYS1.SVCLIB and SYS1.LINKLIB. However, SYSGEN may impose requirements on the contents of these libraries. You should carefully review the section titled "Generation System Requirement" in the SRL "OS/360 System Generation". If any Component Library data set which is recataloged is needed for other jobs during the SYSGEN then you have two options:

a. The other jobs must use volume serials in their JCL to identify the "old" location of the library.

b. You can selectively catalog and uncatalog data sets. This would probably only be useful if you are running the Stage II of SYSGEN as multiple jobs. You could arrange this selective catalog and uncatalog as follows: The jobs for the Stage II would be in a separate job class. When it is desired to start another job the HOLD command would be issued for the SYSGEN class. Then, a job in another class would be initiated to modify the catalog. A job would then be started that would use the modified catalog. When this new job has finished with the catalog, a job would be run to restore the catalog for SYSGEN use. Then the job queue for the Stage II jobs would be released via the RELEASE command.

You can see that these changes in your catalog, together with the fact that the Stage II job stream uses "DISP=OLD" for system data sets, limit the amount of multiprogramming that you can achieve. The Data Set Integrity Feature of MVT will prevent any other job that might use any of these libraries from running at the same time as the SYSGEN.

If you use a Stage I input deck that worked with the Starter System then you may get a B37 ABEND during Stage I. You can avoid this ABEND by putting a space allocation in your Sysprint card for Stage I. A value of (100, 10) in 2311 tracks should be adequate.

Space allocation should also be considered for Stage II. You should review the reader procedure PARM fields for the reader that will be used to read the Stage II jobstream. As above, a value of (100, 10) in 2311 tracks seems to be adequate.

You may want the new system and the driver system to have the same volume serials. You can achieve this result as follows: use IEHDASDR to set the volume serials of the target system to volume serials that are <u>different</u> from the driver. This should be done when these volumes are initialized. When SYSGEN is complete, you can use IEHDASDR to set the volume serials of the new system to the desired values.

A final consideration relates to the use of the Proclib Macro. If you use this Macro you will get the SYS1.PROCLIB that is in the <u>driver</u> system copied into the new system. If you want the PID

procedures or some other procedures, do not use the Proclib Macro; instead you should do a MOVE/COPY after SYSGEN

Stage II as Multiple Jobs

As mentioned above, one way of doing the SYSGEN is to break the Stage II into Multiple Jobs. This allows you to specify parameters, like a region value, in the job cards that you insert in the job stream. In addition, the speed of the Stage II should be improved because the printing of the output of earlier steps can proceed concurrently with the execution of later steps. There are two ways that you can insert the job cards into the Stage II job stream. You can punch out the job stream and insert the cards by hand or you can produce the job stream on tape and use a program to update it. The following exhibit shows sample PL/1 program that will do a tape to tape update of the Stage II jobstream. This program is meant as an example.

Stage II as a Single Job

You may want to run the Stage II jobstream as a single job. In this case, there are some things that you must consider in addition to the points listed above as "General Considerations". You will have to specify a minimum region size which is large enough to run the step having the largest requirements. You can do this by specifying "MIN=200" in your reply to the "Specify System Parameters" message at IPL time. You should also format the Queue when you IPL . You must reserve a large number of 176 byte records for the initiator. A reply of R 00', 750' to the "Specify Jobqueue Parameters" message is suggested.

OS/360 SYSGEN UNDER MVT

The following PL/I program is meant as a sample of the type of program you will need to do a tape to tape update of the Stage I output. This program makes each step of Stage II a separate job. Note that the parameters to be placed on the Job cards are on a card by themselves.

```
//INSERT JOB MSGLEVEL=1
// EXEC PL1LFCLG
//PL1L.SYSIN DD *
 JOBCD:PROC OPTIONS(MAIN);
 DCL
 JB CHAR(7),
 JOBCARD CHAR(71),
 SG CHAR(4),
 NUM CHAR(2),
 CARD CHAR(74);
 ON ENDFILE (SYSUT1) GO TO END;
      JB='//JOBSG';
                                                    JOBCARD='
   JOB MSGLEVEL=1,REGION=100K,PRTY=12,CLASS=D
 ';
 READ:
 GET FILE(SYSUT1)EDIT(SG,NUM,CARD)(A(4),A(2),A(74));
 IF SG='//SG' THEN DO;
 PUT FILE(SYSUT2) EDIT(JB,NUM,JOBCARD)(A);
 END;
 PUT FILE(SYSUT2)EDIT(SG,NUM,CARD)(A);
 GO TO READ;
 END:END;
/*
//GO.SYSUT1 DD UNIT=2400,VOLUME=SER=SYSUT1,LABEL=(,NL),DCB=BLKSIZE=80
//GO.SYSUT2 DD UNIT=2400,VOLUME=SER=SYSUT2,LABEL=(,NL),DCB=BLKSIZE=80
```

EXHIBIT 1

OS/360 MFT RELEASE 17 SCHEDULER SIZE CONSIDERATIONS

Those MFT users planning to use the minimum scheduler designs in Release 17 are reminded that two optional factors, blocking and accounting routine, will affect the scheduler size.

1. The partition size is 30,720 or 45,056 plus buffer space for the Reader/Interpreter. Buffer space is 0 if single buffering and no blocking (Reader INPUT or OUTPUT, and PROCLIB) is used. If the INPUT or PROCLIB data sets are blocked or buffered greater than 1 the storage requirement is increased. If the Reader/Interpreter output data set is blocked or buffered greater than 2 the minimum requirement is increased. Consult the Storage Estimates SRL (C28-6551-7) Page 33 for details.

2. With Release 17 the accounting data set writer IEFWAD is fully supported. This enables an installation accounting routine to record accounting information in the SYS1.ACCT data set from the Initiator/Terminator accounting routine exits.

   If ACCTRTN=SUPPLIED is coded in the SCHEDULR macro at system generation time, the data set writer IEFWAD will be included in the appropriate scheduler modules. IEFWAD occupies 2750 bytes of storage. This figure should be added to the size of the Initiator/Terminator.

Those users who want accounting, but do not require the services of IEFWAD, may modify the Stage I output of SYSGEN in the following manner: locate the INCLUDE cards containing IEFWAD and remove the name IEFWAD from each of them.

OS/360 PASSWORD PROTECTION FOR SYSTEM DATA SETS

Because the accidental or malicious deletion or alteration of system data sets directly impacts the availability and reliability of the system, and because the expiration date protection feature is easily circumvented, it is a worthwhile precaution to password protect system data sets. The password protection feature provides reliable protection against accidental deletion or alteration. It is also effective against the easier methods which the malicious user might employ to damage system data sets, but should not, however, be considered completely reliable in preventing malicious damage.

When a data set is password protected, any attempt to OPEN the data set, or perform such functions as SCRATCH, or RENAME, results in a message to the operator's console which requests the password for the data set. If the job attempting to modify the data set is authorized and has supplied the operator with the correct password, he can reply with the password and the job will be allowed to continue. To avoid needlessly terminating a job because of a careless reply by the operator, two attempts to supply the correct password are allowed. If the password is not correct the second time, the job is abnormally terminated. (See the Messages and Codes publication, message number IEC301A, for a detailed description of the message and operator action.)

To password protect system data sets the following must be considered:

1. If password protection is to be used for system data sets, the expiration date feature need not also be used. For this and other reasons discussed below, the decision to use the password feature for system data sets should be made before system generation.

2. In order to use the password feature, the user must write his own routines to create and maintain a sequential data set name PASSWORD, which associates passwords with data set names and is referenced by the system security routines to determine whether operator-supplied passwords are correct or not. The PASSWORD data set must reside on the system IPL volume and therefore should have space allocated before system generation, at the same time as other system data sets. The PASSWORD data set on the system being generated is not used during system generation, however, and need not contain valid information until the generated system is IPL'd. A detailed description of the PASSWORD data set and the password protection feature can be found in the section "Implementing Data Set Protection" in the System Programmer's Guide. This section should be reviewed in detail before attempting to implement and use the password feature.

3. It is also necessary for the user to create a PASSWORD data set on the IPL volume of the generating system (if one does not exist already). This data set should contain passwords for each of the system data sets on the system to be generated. These passwords need not be the same as those specified in the PASSWORD data set of the system to be generated.

   Data sets to be password protected must have the password option specified when they are first created. (See the description of the LABEL parameter in the Job Control Language publication for information on how to specify this option.) This restriction impacts the system generation process in that if the system data sets on the system to be generated are password protected when they are created, the operator will have to supply a password each time one of the data sets is opened during the system generation process. Such a burden on the operator would generally be considered unacceptable. Two methods of alleviating the burden are proposed:

   a. Circumvent the need to specify password protection when the data sets for the generated system are first allocated. To accomplish this, first initialize an extra set of direct access volumes for the system to be generated. Before system generation, initialize the system data sets on both sets of volumes in exactly the same way except that on one set all system data sets should specify password protection and on the other set no protection should be specified. The unprotected set of volumes should be used for system generation.

      After the system is generated, the generating system can be used to move each of the system data sets (see IEHMOVE program description in the Utilities publication) into its preallocated, and password protected, counterpart on the other set of volumes. This method has the advantages that it only requires the specification of a password once for each system data set (during the move operation) and does not interfere at all with the system generation process. (The passwords requested during the move will be those contained in the PASSWORD data set on the generating system.)

   b. A second, less acceptable method is to specify, in the PASSWORD data set of the generating system, passwords of 'u' for all the system data sets of the system to be generated. This method does not decrease the number of extra operator messages and replies but it does make all replies equivalent to the standard override REPLY 'u'. Caution should be used if this method is used and the system data sets of the generating system are also password protected; since 'u' will also be a valid password for those data sets until the 'u' entries are removed from the PASSWORD data set.

OS/360 INDIRECT CATALOG REFERENCE

The following contribution from England has not been submitted to any formal IBM test. Potential users should evaluate its usefulness in their own environment prior to implementation

A customer has several IS data sets which have their primes and indices on different volumes. It was required that these data sets were recreated on a regular basis in order to reorganize them.

Space problems prohibited writing a new prime on the same disk as the old prime, so the new prime had to go where the old index was located, and similarly, the new index on the volume containing the old prime.

This meant that the location of the prime and index was variable, being one of two disks in each case.

Generation data groups were considered, but soon abandoned as a specification of DSNAME=ISDATA (INDEX)(+1) is not supported.

Also the JCL manual, C28-6539, states: "An ISAM data set can be cataloged using CATLG" (meaning in the DISP parameter of DD statement) "Only if all three areas are defined by the same DD statement". This is mentioned in Appendix D: Creating and retrieving indexed sequential data sets.

The method adopted was therefore:
Consider the IS data set called ISDATA.

Two catalog entries were created, viz, ISDATAI and ISDATAP, pointing to the

volumes containing the index and prime respectively.

Assume that the catalog entries are correct, then the following JCL coding will work every time it is run.

```
//          EXEC   PGM=.....
// OLD      DD     DSN=ISDATA, VOL=
                   REF=ISDATAI, DISP=
                   (OLD,DELETE, KEEP)
//          DD     DSN=ISDATA, VOL=
                   REF=ISDATAP, DISP=
                   (OLD, DELETE, KEEP)
// NEW      DD     DSN=NEW, VOL=REF=
                   ISDATAP, DISP=(NEW,
                   KEEP, DELETE), etc.
//          DD     DSN=NEW, VOL=REF=
                   ISDATAI, DISP=(NEW,
                   KEEP, DELETE), etc.
// OLDI     DD     DSN=ISDATAI, DISP=
                   (OLD, UNCATLG
                   KEEP)
// OLDP     DD     DSN=ISDATAP, DISP=
                   (OLD, UNCATLG,
                   KEEP)
// NEWI     DD     DSN=ISDATAI, VOL=
                   REF=ISDATAP, DISP=
                   (OLD, CATLG, KEEP)
// NEWP     DD     DSN=ISDATAP, VOL=
                   REF=ISDATAI, DISP=
                   (OLD, CATLG, KEEP)
```

After an unsuccessful execution of the above, i.e. after an ABEND, everything will remain unchanged.

After successful execution the data set ISDATA will be deleted and the data set, NEW, will be kept.

It is imperative that the DD statements OLDI and OLDP occur before NEWI and NEWP, as catalog management at step termination is done sequentially starting from the first entry. Were NEWI to occur before OLDI then no change would be made to the catalog as ISDATAI would already be cataloged. Being in the order shown, ISDATAI is first uncataloged and then cataloged on the volume that held the old prime Similarly ISDATAP is switched.

An execution of program IEHPROGM must then follow to rename NEW on both volumes to ISDATA. IEHPROGM will treat an index or a prime as a data set in its own right.

In the event of the catalog entries being wrong then system completion code 03D will occur during OPEN for the IS data set. This fact can be utilized in a program for fixing the catalog. A program that simply opens the IS data set can be run with JCL similar to the above except that on ABEND the catalog entries are switched, and on successful execution they are left unchanged.

Logically, the idea could be extended to cater for IS data sets with an independent overflow area also.

OS/360 RELEASE 17 EXTENDED PARM FIELD

The expansion of the EXEC statement PARM field from 40 to 100 characters as described in the Release 17 Job Control Language manual has been misinterpreted by some readers. Using apostrophes to enclose individual expressions which contain special characters, such as equal signs, is required only when a PARM field must be continued to the next card In this case, parentheses are used to enclose the entire value field.

Those 40 character PARM fields which are acceptable to Release 15/16 are also valid in Release 17.

OS/360 READER FOR PUNCH SIDE OF 2540

The following field contribution has not been submitted to any formal IBM test. Potential users should evaluate its usefulness in their own environment prior to implementation.

PUNCHRDR is a program that reads cards through the punch side (00D) of the 2540 and creates a sequential data set on a direct access device. It was designed to create a job stream on disk in an emergency situation when the read side of the 2540 is down. The program uses EXCP and QSAM to read cards and write records.

The reader procedures accompanying this program allow the operator to initiate this program without the use of the card reader. i. e. The operator starts RDR00D to read cards at the punch (00D), and then starts RDRDISK to read/interpret the job stream on disk.

The program was written in OS Assembler and tested on Release 16 of OS/360.

It has the minimum system requirements of OS/360 MFT/MVT and uses the punch-feed-read (PFR) feature on the 2540 and approximately 2K of memory.

Included in the exhibits on the following three pages is:

1. The program, PUNCHRDR, which should be assembled and link edited into SYS1. LINKLIB.

2. The procedures, RDR00D, RDRDISK, and RDR00DEW, which should be placed into SYS1. PROCLIB.

OS/360 DATA CELL SUPPORT CLARIFICATION

OS/360 supports 2321 load module libraries when the 2321 is attached to a selector channel. Furthermore, support exists for these libraries on 2870 selector subchannels where PCI FETCH is not included in the system. PCI FETCH can be optioned out in PCP and MFT. MVT does not support 2321 Program Libraries when the 2321 is attached to a selector subchannel.

With the availabity of the LOADER, an alternate plan for installations with 2321's on the 2870 selector subchannels might be to investigate the applicability of the use of the OS/360 LOADER facilities to do a "compile load/go" or a "load/go" directly from 2321.

The sales manual pages are being changed to reflect this level of support.

OS/360 RELEASE 17 CONSOLIDATED DOCUMENT CORRECTION

On page 1-21 of the Consolidated Document for Release 17 under the heading SYSGEN Macro Instructions, the data management macro description reads:

This macro instruction must not be used if SVC-2B (the CIRB macro instruction) is to be included in the generated system.

This description should read:

This macro instruction must now be used if SVC-2B (the CIRB macro instruction) is to be included in the generated system.

OS/360 READER FOR PUNCH SIDE OF 2540

```
STMT   SOURCE STATEMENT
   1 PUNCHRDR START
   2          PRINT   NOGEN
   3          SAVEREG (this macro generates save logic)
  20          WTO     'THE PUNCH-FEED-READ FEATURE ON THE 2540 CAN NOT READ X
                      CARDS PUNCHED IN COLUMN BINARY(N)   I.E. OBJECT DECKS CANX
                      NOT BE READ'                  →11,5,9 punch
  29          WTO     'A CARD IN POCKET 1 MEANS THAT THERE WAS A READ ERROR A
                      AND THAT CARD HAS NOT BEEN READ'
  38 OPEN     EQU     *
  39          OPEN    (DCBPUNCH,,DCBDISK,(OUTPUT))

  48 READ     EQU     *
  49          OI      DCBPUNCH+44,X'0C'
  50          MVI     ECBPUNCH,X'00'
  51          EXCP    IOBPUNCH
  54          WAIT    ECB=ECBPUNCH
  58          CLI     ECBPUNCH,X'7F'
  59          BE      WRITE
  60          B       ERROR

  62 WRITE    EQU     *
  63          PUT     DCBDISK,AREAOUT
  68          MVC     AREAOUT,AREAIN
  69          B       READ
  70 ERROR    EQU     *
  71          CLI     ECBPUNCH,X'44'
  72          BE      PUNCHERR
  73          CLI     ECBPUNCH,X'41'
  74          BNE     READERR
  75          CLI     CSW+4,X'01'
  76          BNE     READERR
  77          B       EOF
  78 READERR  EQU     *
  79          MVI     WTORECB,X'00'
  80          WTOR    'INTERVENTION REQUIRED ON 00D -- READ ERROR(N)   11,5,9 punch  PRES
                      SS END OF FILE(N)      REPLY ''S'' TO START READER AGAIN',X
                      ANS,1,WTORECB  → 11,5,9 punch
  92          WAIT    ECB=WTORECB
  96          OI      ANS,X'40'
  97          CLI     ANS,C'S'
  98          BE      READ
  99          B       READERR
 100 PUNCHERR EQU     *
 101          MVI     WTORECB,X'00'
 102          WTOR    'INTERVENTION REQUIRED ON 00D -- LAST CARD HAD A VERIFI
                      ICATION CHECK(N)      REPLY ''S'' TO START READER AGAIN',X
                      ANS,1,WTORECB  11,5,9 punch
 114          WAIT    ECB=WTORECB
 118          OI      ANS,X'40'
 119          CLI     ANS,C'S'
 120          BE      READ
 121          B       PUNCHERR
```

EXHIBIT 1

OS/360 READER FOR PUNCH SIDE OF 2540

```
STMT   SOURCE STATEMENT
 123 EOF       EQU     *
 124           MVC     CCWADD,CCWEOFAD
 125           NI      DCBPUNCH+44,X'3F'
 126           MVI     ECBPUNCH,X'00'
 127           EXCP    IOBPUNCH
 130           WAIT    ECB=ECBPUNCH
 134           PUT     DCBDISK,AREAOUT
 139 END       CLOSE   (DCBPUNCH,,DCBDISK)                    11,5,9 punch
 147           WTO     'A JOB STREAM NOW EXIST ON A DIRECT ACCESS DEVICE(N)  X
                     START ''RDRDISK'' TO READ/INTERPRET THIS JOB STREAM'
 156           RETURNX (this macro generates return logic)

 164 DCBPUNCH DCB      DDNAME=PUNCH,DEVD=PC,DSORG=PS,MACRF=E,RECFM=F
 165                 *,***  IHB063  DDNAME SHORT-PADDED TO 8 CHAR

 200 DCBDISK  DCB      DDNAME=DISK,DSORG=PS,MACRF=PM,OPTCD=WC
 201                 *,***  IHB063  DDNAME SHORT-PADDED TO 8 CHAR

 258 IOBPUNCH DS       0F
 259          DC       XL1'42',XL3'0'
 260          DC       A(ECBPUNCH)
 261 CSW      DC       2F'0'
 262 CCWADD   DC       A(CCWREAD)
 263          DC       A(DCBPUNCH)
 264          DC       2F'0'

 266 CCWREAD  CCW      X'C2',AREAIN,X'40',80
 267 CCWPFR   CCW      X'49',BLANK,0,80
 268 CCWSTACK CCW      X'41',BLANK,0,80

 270 AREAIN   DC       CL80'//'
 271 AREAOUT  DC       CL80'//'
 272 BLANK    DC       CL80' '
 273 ECBPUNCH DC       F'0'
 274 WTORECB  DC       F'0'
 275 CCWEOFAD DC       A(CCWSTACK)
 276 ANS      DC       C'F'
 277          END      PUNCHRDR
```

EXHIBIT 1 (Continued)

OS/360 READER FOR PUNCH SIDE OF 2540

```
SYSIN                                   NEW MASTER

./ ADD NAME=PUNCHRDR,LIST=ALL
                                        //IEFPROC EXEC PGM=IEFIRC,REGION=44K,
                                        //            PARM='80103005001024905010SYSDA
                                        //IEFRDER DD DSN=SYS1.PROCLIB(RDROODEW),DISP=SHR,
                                        //           DCB=(BLKSIZE=80,RECFM=F,BUFL=80,BUFNO=1)
                                        //IEFPDSI DD DSN=SYS1.PROCLIB,DISP=SHR
                                        //IEFDATA DD UNIT=SYSDA,SPACE=(80,(5,5),RLSE,CONTIG),
                                        //           DCB=(BUFNO=2,LRECL=80,BLKSIZE=80,RECFM=F,BUFL=80)
./ ALIAS NAME=RDROOD
IEB516I MEMBER NAME (PUNCHRDR) FOUND IN NM DIRECTORY. TTR IS NOW ALTERED.
IEB516I MEMBER NAME (RDROOD  ) FOUND IN NM DIRECTORY. TTR IS NOW ALTERED.

SYSIN                                   NEW MASTER

./ ADD NAME=RDROODEW,LIST=ALL
                                        //PUNCHRDR JOB ,'E.E.WERLE',MSGLEVEL=1,REGION=12K,PRTY=14
                                        // EXEC PGM=PUNCHRDR
                                        //DISK    DD UNIT=SYSDA,DSN=JOBSTREM,DCB=(BLKSIZE=80,RECFM=F,BUFL=80,
                                        //           BUFNO=1),SPACE=(TRK,(20,20)),DISP=(NEW,CATLG,DELETE)
                                        //PUNCH DD UNIT=OOD
                                        //SYSUDUMP DD SYSOUT=A
./ ENDUP
IEB516I MEMBER NAME (RDROODEW) FOUND IN NM DIRECTORY. TTR IS NOW ALTERED.

SYSIN                                   NEW MASTER

./ ADD NAME=RDRDISK,LIST=ALL
                                        //IEFPROC EXEC PGM=IEFIRC,REGION=44K,
                                        //            PARM='80103005001024905010SYSDA
                                        //IEFRDER DD          DSN=JOBSTREM,DCB=(BLKSIZE=80,RECFM=F,BUFL=80,
                                        //           BUFNO=1),DISP=(OLD,DELETE)
                                        //IEFPDSI DD DSN=SYS1.PROCLIB,DISP=SHR
                                        //IEFDATA DD UNIT=SYSDA,SPACE=(80,(500,500),RLSE,CONTIG),
                                        //           DCB=(BUFNO=2,LRECL=80,BLKSIZE=80,RECFM=F,BUFL=80)
IEB516I MEMBER NAME (RDRDISK ) FOUND IN NM DIRECTORY. TTR IS NOW ALTERED.
```

EXHIBIT 2

DOS/360 POWER POINTERS

1. The user can specify when he generates his POWER system that he wants a reader (or a writer in a writer-only system) started automatically. The specific macro parameter is AUTOSTR=nnn, where nnn is the address of the card reader or printer to be started when the system is initiated. If the parameter is coded AUTOSTR=NO the system operator must start the reader task by using the Start command.

   The obvious advantage of using the AUTOSTART facility is that it reduces operator effort and helps bring up the system faster. The disadvantage is a little less obvious. If the reader (or printer) is down and you've specified AUTOSTART you cannot bring up the system.

   While a system with one card reader is probably down when the reader is down, users of writer-only systems might want to forego the operational ease of AUTOSTART for the ability to run with a "tape writer" should the printer go down. People with more than one card reader should think carefully about specifying AUTOSTART for one of the readers as they would not be able to run with POWER if that reader were inoperative.

   Code has been developed to extend the AUTOSTART facility to enable a user with both a reader and a writer to have both tasks started automatically. These modifications are available from your local Field Systems Center POWER coordinator.

2. Autobuffers

   After a reader or a writer has been initiated the system asks the operator how many buffers are to be allocated to this task. The only acceptable responses are 1 or 2.

   Modifications have been made to the POWER system to make the buffer assignment a constant decided at SYSGEN time. This eliminates one more series of operator responses. The modifications are available from your local Field Systems Center.

3. Reader Close

   At POWER generation time the user can specify one of three options for Reader close-PAUSE, GO or EXIT

   a. RDRCLOS=PAUSE

   This is the default option in the POWER macro. After the last card of the last job is read in the system types BGROOC CLOSED and the reader task is made inactive. The buffers assigned to the reader are not freed. To restart the reader task the operator uses the GO command not the Start command. (Start is used to initiate another task, reader or writer).

   b RDRCLOS=GO

   A standard DOS intervention required message is issued after the last card is read. The task becomes inactive pending an I/O interrupt from the card reader. Restarting the task is simplicity itself-put cards in the card reader and press start.

   c. RDRCLOS=EXIT

   At end-of-file and end-of-job the system types BGROOC CLOSED as with the RDRCLOSE=PAUSE option. However, the buffers are freed and the reader task exits core. When it becomes necessary to read cards again a new reader must be initiated using the Start command.

4. Freeing Buffers

   If you've coded RDRCLOS=EXIT the buffers are automatically free at end-of-file. If you've coded PAUSE or GO the operator can Cancel or Stop a reader or a writer and make the buffers available.

   a. Active task

   When an active reader or writer is Stopped the task is deactivated and the buffers are freed. Cancel has the same effect on a reader task. However, Cancel directed to a writer deletes the current job before deactivating the task.

b. Inactive task

The Stop and Cancel commands do not directly deactivate the tasks. The command processor routine simply posts a task control bit and the task subsequently deactivates itself. This means that if you Stop or Cancel an inactive task you must then re-activate it in order to make the buffers available.

For a system coded with RDRCLOSE=GO you have to put a few blank cards in the reader and start up the reader task. For a system coded with RDRCLOSE=PAUSE the operator must issue a Go command after the Stop or Cancel in order to have the reader detach itself.

S/360 1401 COMPATIBILITY DIRECT SEEK

This field contribution concerns 1401 Emulation on the 360/25 and 1401 Compatibility on the 360/30.

When the 1400 Autocoder library resides on a separate pack, specifying the "Direct-Seek" option on a library update causes the drive to seek to cylinder 200 and hang up when cylinder 0 seek is issued by the update program. Program Halt 4 (drive not ready) occurs. Eliminate this problem by not specifying "Direct Seek" in the option card. There is no lost time as all 360-2311 seeks are direct.

STAND ALONE EMULATOR WITH COS 50 RPQ

This field contribution has not been submitted to any formal IBM test.

One of the functions of the COS 50 RPQ is to allow certain tables, which were required to be located at address X'2000' by the Stand Alone Emulator, to be relocated. This is accomplished by loading the address of the tables into FPR2. The compatibility feature with the RPQ then uses FPR2 as a base pointer for a reference to the tables.

In order to run the Stand Alone Emulator after the COS 50 RPQ has been installed, the user must modify symbolic location LNKOUT-4 to X'00002000', using a standard .REP card.

BASIC PROGRAM PACKAGE
DOCUMENTATION - Write-up.
MACHINE READABLE - Source code and sample problem.

OPTIONAL PROGRAM PACKAGE - None.

ORDERING INFORMATION: PROGRAM NUMBER 360D005005

	PROGRAM NUMBER EXTENSION	DISTRIBUTION MEDIUM TYPE	CODE	USER VOLUME REQUIREMENT
BASIC	none	DTR*	00	none
OPTIONAL	none	none		none

* * * * * * * * NEW TYPE III PROGRAM * * * * * * * *

EQUIPMENT ORDER, SHIPMENT, INSTALLATION, AND DISCONTINUANCE REPORT FOR LARGE ACCOUNTS

DESCRIPTION - A general purpose program to produce a variable, 24 month, graphic representation of on-order and installed equipment. This program is useful (particularly for large accounts) to both IBM representatives and customers to tell at a glance what equipment is scheduled to be shipped, installed, or removed at different physical locations. Three levels of totals are maintained for the 24 month period under consideration at any particular execution. These are (1) Totals for each physical location (2) Totals for each logical grouping of locations (3) Grand totals for all locations. Input is based on a simple detail card with default values available wherever possible.

PROGRAMMING SYSTEMS - Written in OS/PL/I showing examples of various PL/I coding techniques. Tested under Release 15/16 of the Operating System using Version IV.

MINIMUM SYSTEM REQUIREMENTS - An OS/360 of PL/I 64K execution, region or partition.

BASIC PROGRAM PACKAGE
DOCUMENTATION - Write-up.
MACHINE READABLE - Source code and sample problem.

OPTIONAL PROGRAM PACKAGE - None.
Ordering Information: Ordering Information: Program Number 360D-25.0.004

	PROGRAM NUMBER EXTENSION	DISTRIBUTION MEDIUM TYPE	CODE	USER VOLUME REQUIREMENT
BASIC	none	DTR 9/800	28	none
		DTR 9/1600	29	none
		DTR 7DC/800	26	none
OPTIONAL	none	none		none

* * * * * * * * NEW TYPE III PROGRAM * * * * * * * *

S/360 DOS TAPE TO PRINTER/PUNCH SPOOLING SYSTEM

DESCRIPTION - Program prints/punches data from magnetic tape created for spooling. It accepts output from DOS SYSOUT, OS SYSOUT, COS and all programming languages. Output is identical to "On-Line" printing. Handles intermixed blocked/unblocked fixed/variable length printer and/or punch records. Handles multi-job, multi-file and multi-volume labeled and unlabeled tapes. Prints multiple copies. Provides for forms changes; flip-flops multiple input tapes. Accounting routine punches cards containing job name, date, elapsed job and spool time, number of input/output records, operator and machine identification. Very significant

CONTRIBUTED PROGRAMS

The following are the abstracts of new Type III and Type IV programs and the description of revisions to existing Type III and Type IV programs available from the Program Information Department. Complete program abstracts and revisions will be incorporated into a future issue of the IBM Catalog of Programs.

Type III (IBM Contributed Programs) and Type IV (Customer Contributed Programs) are programs of general interest contributed to the Program Information Department for distribution. These programs and their related documentation are essentially in the author's original form and have not been subjected to any formal testing. IBM makes no warranty, expressed or implied, as to the documentation, function or performance of these programs and the user is expected to make the final evaluation as to their usefulness in his own environment. There is no committed maintenance for these programs. However, any changes the author chooses to make will be announced in subsequent issues of the IBM Installation Newsletters and the Catalogs of Programs.

Programs may be obtained by submitting a properly completed "IBM Program Order" form (Form Number 120-1957) to the Program Information Department, 40 Saw Mill River Road, Hawthorne, New York 10532.

* * * * * * * * * REVISION * * * * * * * * *

360E-21.1.002 COMPLETE PAYROLL APPLICATION FOR 360/20 CARD SYSTEM. Material Revised - The documentation, machine readable, and the catalog abstract have been changed. (Source code is now basic material. Object code is no longer available).

BASIC PROGRAM PACKAGE
DOCUMENTATION - Write-up.
MACHINE READABLE - Source code and sample problem.

OPTIONAL PROGRAM PACKAGE - None.

ORDERING INFORMATION: PROGRAM NUMBER 360E211002

	PROGRAM NUMBER EXTENSION	DISTRIBUTION MEDIUM TYPE	CODE	USER VOLUME REQUIREMENT
BASIC	none	DTR*	00	none
OPTIONAL	none	none		none

* * * * * * * * NEW TYPE III PROGRAM * * * * * * *

FORMATTED LISTING OF VTOC FOR OS/360

DESCRIPTION - This program provides the user with an easily readable listing of the VTOC of an OS/360 2311 or 2314 volume. All numeric information is in decimal and identified by keywords or headings. The information provided includes data set name, file organization, extents, total current space allocation, allocation type (CYL, TRK, etc.), secondary allocation quantity, creation and expiration dates, and DCB information. For partitioned and sequential files, the number of tracks used is given. For fixed length-record sequential files, the number of records is calculated. Selected statistical information is provided for indexed sequential data sets. A volume summary provides a list of the size of all free extents and the total free area. The program also provides an optional link of IEHLIST to produce catalog and PDS directory listings in the same step.

PROGRAMMING SYSTEMS - Written in PL/1 and BAL.

MINIMUM SYSTEM REQUIREMENTS - An OS/360 system with PL/1 and a 60K region.

increase in thruput. One program handles all spooled output, provides accounting information, eliminates special tape and program handling.

PROGRAMMING SYSTEMS - Written in DOS Assembler.

MINIMUM SYSTEM REQUIREMENTS - 1 magnetic tape drive, 2540, 1403 console. Read/write channel switch desirable. Uses 6K to 10K partition.

BASIC PROGRAM PACKAGE
DOCUMENTATION - Write-up.
MACHINE READABLE - Object code, source code and sample problem.

OPTIONAL PROGRAM PACKAGE - None.

ORDERING INFORMATION: PROGRAM NUMBER 360D003025

	PROGRAM NUMBER EXTENSION	DISTRIBUTION MEDIUM TYPE	CODE	USER VOLUME REQUIREMENT
BASIC	none	DTR 9/800	28	none
		DTR 9/1600	29	none
		DTR 7DC/800	26	none
OPTIONAL	none	none		none

* * * * * * * * NEW TYPE III PROGRAM * * * * * * *

DOS IOCS LOGIC MODULE SOURCE STATEMENT GENERATOR

DESCRIPTION - Given the DOS generated standard IOCS logic module name (e.g. IJCFCIZO), this program will produce, on SYSPCH and SYSLST, the DOS job control cards and source statements (macro keywords and operands), in the proper sequence, to assemble the module. The program is self-relocating. Input is from SYSLOG or SYSIPT, where SYSIPT is a card reader, disk or tape drive supported by DOS. All module names submitted are checked for validity, and due to the ability to produce a complete job stream of many assemblies in one run, the program could significantly reduce desk time and keypunching at system generation time.

PROGRAMMING SYSTEMS - Written in Basic Assembler Language, and will run on any system capable of supporting DOS.

MINIMUM SYSTEM REQUIREMENTS - Same as those required for DOS/360.

BASIC PROGRAM PACKAGE
DOCUMENTATION - Write-up.
MACHINE READABLE - Object code, source code and sample problem.

OPTIONAL PROGRAM PACKAGE - None.

ORDERING INFORMATION: PROGRAM NUMBER 360D030008

	PROGRAM NUMBER EXTENSION	DISTRIBUTION MEDIUM TYPE	CODE	USER VOLUME REQUIREMENT
BASIC	none	DTR*	00	none
OPTIONAL	none	none		none

* * * * * * * * NEW TYPE III PROGRAM * * * * * * *

PROOF OF DEPOSIT ENTRY RUN WITH TRANSIT DISTRIBUTION

DESCRIPTION - This is a generalized MICR capture and control system designed to provide a bank with a flexible means of producing proof of deposit, transit distribution, cash letters, float analysis, batch proof, and transactions capture. The program supports multi-bank processing and interfaces with the Demand Deposit Accounting Program, Program Order No. 360D-19.7.003. To implement the program the user completes a set of decision tables. The design and use of these tables provide a simplified approach to the initial definition and subsequent maintenance of the application. These tables are used to uniquely identify each document and thus to determine its proper processing and distribution pattern. The program accepts both MICR documents and punched cards as input. Printed output consists of batch lists, block totals, processing recaps, and cash letters.

PROGRAMMING SYSTEMS - Written in DOS Assembly Language.

MINIMUM SYSTEM REQUIREMENTS - A S/360 Model 25E, one 2311, and a 1419 or 1412. The program runs under 10K DOS with MICR support.

BASIC PROGRAM PACKAGE
DOCUMENTATION - Write-up.
MACHINE READABLE - Appropriate material delivered.

OPTIONAL PROGRAM PACKAGE - None.

ORDERING INFORMATION: PROGRAM NUMBER 360D197010

	PROGRAM NUMBER EXTENSION	DISTRIBUTION MEDIUM TYPE	CODE	USER VOLUME REQUIREMENT
BASIC	none	DTR 9/800	28	none
		DTR 9/1600	29	none
		DTR 7DC/800	26	none
OPTIONAL	none	none		none

* * * * * * * * NEW TYPE III PROGRAM * * * * * * * *

S/360 MOD 44 MULTIPROGRAMMING SYSTEM

DESCRIPTION - This program supports external interrupt under DAMPS Version 2 (360A-CX-20X), eliminating the need for the Priority Interrupt special feature. Consequently, DAMPS will execute on a standard S/360 or a Model 44 with special features. The resulting system supports two independent, storage protected partitions. The background partition supports a standard Mod 44 P/S job stream. The foreground partition assigns program tasks to six levels of external interrupts, the interrupt key, or dynamically schedules them for multiprogrammed execution while external interrupts are inactive. For on-line scientific application, DAMPS supports a real-time interface for OEM devices. With the exception that the Priority Interrupt, External Interrupt and Memory Protection special features are not required.

PROGRAMMING SYSTEMS - Source language is BAL.

MINIMUM SYSTEM REQUIREMENTS - Same as for DAMPS, Version 2.

BASIC PROGRAM PACKAGE
DOCUMENTATION - Write-up.
MACHINE READABLE - Source code and sample problem.

OPTIONAL PROGRAM PACKAGE - None.

ORDERING INFORMATION: PROGRAM NUMBER 360D030007

PROGRAM NUMBER EXTENSION	DISTRIBUTION MEDIUM TYPE	CODE	USER VOLUME REQUIREMENT

	PROGRAM NUMBER EXTENSION	DISTRIBUTION MEDIUM TYPE		CODE	USER VOLUME REQUIREMENT
BASIC	none	DTR	9/800	28	none
		DTR	9/1600	29	none
		DTR	7DC/800	26	none
OPTIONAL	none	none			none

* * * * * * * * NEW TYPE III PROGRAM * * * * * * *

UNIT CONTROL SYSTEM FOR RETAILERS

DESCRIPTION - A complete set of System/360 programs is provided for the small to medium size retailer to collect and maintain sales and on-hand information. After the transaction (sales and receipts of merchandise, merchandise transfers, returns to vendors, inventory adjustments, etc.), they are machine-edited and checked for validity, written on a disk file, and sorted into sequence. The master records contain sales and on-hand information to the department, class, price, resource, style, and color/size level. The updated records are processed at appropriate intervals to prepare reports for the buyers and merchandise management. The reports supplant the information previously available in the style black books, and provide additional exception and analysis information.

PROGRAMMING SYSTEMS - Written in Assembly Language and run under the Basic Operating System (BOS).

MINIMUM SYSTEM REQUIREMENTS - Model 25 with 2 disk drives and 16K storage.

BASIC PROGRAM PACKAGE
DOCUMENTATION - Write-up.
MACHINE READABLE - Object code, source code and sample problem.

OPTIONAL PROGRAM PACKAGE - None.

ORDERING INFORMATION: PROGRAM NUMBER 360D252004

	PROGRAM NUMBER EXTENSION	DISTRIBUTION MEDIUM TYPE		CODE	USER VOLUME REQUIREMENT
BASIC	none	MT	9/800	28	01
		MT	9/1600	29	01
		MT	7DC/800	26	01
OPTIONAL	none	none			none

* * * * * * * * NEW TYPE III PROGRAM * * * * * * *

JULIAN DATA PROCESSING SUBROUTINES

DESCRIPTION - This submission consists of four subroutines. The subroutines will (1) Convert a pack Julian date to a packed MMDDYY form (2) Convert a packed MMDDYY date to a packed Julian Date, (3) Calculate the difference between two Julian dates and (4) Extract the DPS system date in both MMDDYY and Julian form. All dates are processed packed decimal.

PROGRAMMING SYSTEMS - Written in Model 20 DPS Assembler Language and designed for use with Model 20 DPS RPG.

MINIMUM SYSTEM REQUIREMENTS - Same as those required for 360 Model 20 DPS.

BASIC PROGRAM PACKAGE
DOCUMENTATION - Write-up.
MACHINE READABLE - Object code, source code and sample problem.

OPTIONAL PROGRAM PACKAGE - None.

ORDERING INFORMATION: PROGRAM NUMBER 360E065002

	PROGRAM NUMBER EXTENSION	DISTRIBUTION MEDIUM TYPE	CODE	USER VOLUME REQUIREMENT
BASIC	none	DTR*	00	none
OPTIONAL	none	none		none

* * * * * * * * NEW TYPE III PROGRAM * * * * * * *

1130 PATIENT BILLING

DESCRIPTION - The program accepts patient charges and provides for patient insurance company billing. The package covers admissions, charge posting, census reporting, insurance proration, preparation of detail and summary bills. Also included are updating of room transfers and discharges. The program provides Medicare Billing for Part A (SSA 1452C) and Part B (SSA 1554).

PROGRAMMING SYSTEMS - Written in 1130 FORTRAN utilizing 1130 Commercial Subroutines.

MINIMUM SYSTEM REQUIREMENTS - 1131 Model 2B (console printer and keyboard) 8K, #3854 Expansion Adapter, #3616 1132 Printer Attachment, #4454 1442 Card Read-Punch Attachment, 1132 Printer Model 1, 1442 Card Read-Punch Model 6, 2310 Disk Storage Model B1, 029 Card Punch.

BASIC PROGRAM PACKAGE
DOCUMENTATION - Write-up.
MACHINE READABLE - Object code, source code and sample problem.

OPTIONAL PROGRAM PACKAGE - None.

ORDERING INFORMATION: PROGRAM NUMBER 1130282002

	PROGRAM NUMBER EXTENSION	DISTRIBUTION MEDIUM TYPE	CODE	USER VOLUME REQUIREMENT
BASIC	none	CARDS	15	none
OPTIONAL	none	none		none

* * * * * * * * NEW TYPE III PROGRAM * * * * * * *

1130 HOSPITAL ACCOUNTS RECEIVABLE

DESCRIPTION - 1130 Hospital Accounts Receivable offers the smaller hospital user the full power of a stored program computer with rapid accessibility to patient accounts receivable files stored on disk. The program provides for updating of patient account files, printing of statements, aged accounts analysis, Account Status, Insurance Accounts Receivable, deleted accounts for zero or small balance. An accounts receivable control log is used to record the number of accounts and dollar value of receivables in the system.

PROGRAMMING SYSTEMS - Written in 1130 FORTRAN utilizing 1130 Commercial Subroutines.

MINIMUM SYSTEM REQUIREMENTS - 1131 Model 2B (console printer and keyboard) 8K, #3854 Expansion Adapter, #3616 1132 Printer Attachment, #4454 1442 Card Read-Punch Attachment, 1132 Printer Model 1, 1442 Card Read-Punch Model 6, 2310 Disk Storage Model B1, 029 Card Punch.

BASIC PROGRAM PACKAGE
DOCUMENTATION - Write-up.
MACHINE READABLE - Object code, source code and sample problem.

OPTIONAL PROGRAM PACKAGE - None.

ORDERING INFORMATION: PROGRAM NUMBER 1130283001

	PROGRAM NUMBER EXTENSION	DISTRIBUTION TYPE	MEDIUM CODE	USER VOLUME REQUIREMENT
BASIC	none	CARDS	15	none
OPTIONAL	none	none		none

* * * * * * * * NEW TYPE IV PROGRAM * * * * * * *

A PROGRAM FOR GRAPHIC ON-LINE DEBUGGING ON AN IBM 2250 DISPLAY

DESCRIPTION - The Graphic On-Line Debugging program (GOLDBUG) is a program designed as an aid to on-line debugging, specifically with regard to programs involving interactive graphics. The basic capabilities offered by GOLDBUG are: (1) Examination of core and the registers. Core is displayed in blocks of 768 bytes in a dump-like format. The area displayed can be changed by scrolling, address computation (including relative addressing), or "pointer chasing", using the light pen to indicate the word containing the next address to be displayed. (2) The ability to patch any part of core. (3) The option of producing hard copy of any part of core currently displayed, or requesting a full-task snap. (4) Fixed and floating point conversion. The word (or words, for double precision) to be translated is indicated with the light pen. (5) The ability to call any in-core subroutine from GOLDBUG with register contents that can be specified by the programmer. GOLDBUG can be used as an attention routine or can be called from an appropriate SPIE routine. In the latter case, it is possible to display all the registers and the PSW as they were when the interrupt occured.

PROGRAMMING SYSTEMS - Written in Assembly Language using GPS.

MINIMUM SYSTEM REQUIREMENTS - An S/360 MOD 40 or higher and equipped with a 2250 MOD 1 or 3 scope.

BASIC PROGRAM PACKAGE
DOCUMENTATION - Write-up.
MACHINE READABLE - Source code.

OPTIONAL PROGRAM PACKAGE - None.

ORDERING INFORMATION: PROGRAM NUMBER 360D046001

	PROGRAM NUMBER EXTENSION	DISTRIBUTION TYPE	MEDIUM CODE	USER VOLUME REQUIREMENT
BASIC	none	DTR 9/800	28	none
		DTR 9/1600	29	none
		DTR 7DC/800	26	none
OPTIONAL	none	none		none

* * * * * * * * NEW TYPE IV PROGRAM * * * * * * *

GENERAL TAXONOMY PROGRAM

DESCRIPTION - This program allocates a set of individuals into groups. Each group has typical individual and the members of the group are all sufficiently similar

to the type. The user defines the similarity needed. The types are determined to carry information of their group members only. The variables are quantitative ones. With some changes in the program qualitative could be used also.

PROGRAMMING SYSTEMS - Written in Basic FORTRAN IB, and has been compiled and tested using TOS Version 11 on a S/360 Model 30. It could be used in other operating systems also. To get a complete understanding of the method it is necessary to read the Write-up.

MINIMUM SYSTEM REQUIREMENTS - Same as those required for a S/360 Model 30.

BASIC PROGRAM PACKAGE
DOCUMENTATION - Write-up.
MACHINE READABLE - Source code and sample problem.

OPTIONAL PROGRAM PACKAGE - None.

ORDERING INFORMATION: PROGRAM NUMBER 360D131005

	PROGRAM NUMBER EXTENSION	DISTRIBUTION MEDIUM TYPE	CODE	USER VOLUME REQUIREMENT
BASIC	none	DTR*	00	none
OPTIONAL	none	none		none

* * * * * * * * NEW TYPE IV PROGRAM * * * * * * *

FIELD SELECTION AND LIST

DESCRIPTION - This program will list card fields (in label form) in the order designated on a control card. The labels are single spaced with a skip to channel 1 after each label. Up to nineteen lines can be selected from each data card. A control card punched as follows would cause columns 41 to 60, 61 to 75, and 01 to 10 to produce three lines of print in that field order: 416061750110. A control card specification of 0180 would cause all eighty columns of the card to produce one line of print, listing the entire card. Fields can be overlapped if required. A control card line specification of 0000 will produce an extra space between label lines. An additional space will be attained for each additional 0000 control card entry.

PROGRAMMING SYSTEMS - Written in BAL.

MINIMUM SYSTEM REQUIREMENTS - This program was written for the System/360 Model 20 using a 2560 Multifunction Card machine and a 2203 Printer. Core storage capacity may be from 4-16K.

BASIC PROGRAM PACKAGE
DOCUMENTATION - Write-up.
MACHINE READABLE - Object and source code.

OPTIONAL PROGRAM PACKAGE - None.

ORDERING INFORMATION: PROGRAM NUMBER 360E010002

	PROGRAM NUMBER EXTENSION	DISTRIBUTION MEDIUM TYPE	CODE	USER VOLUME REQUIREMENT
BASIC	none	DTR*	00	none
OPTIONAL	none	none		none

* * * * * * * * NEW TYPE IV PROGRAM * * * * * * *

MULTIPLE RANGE TEST FOR CORRELATED AND HETEROSCEDASTIC MEANS

DESCRIPTION - This program performs a Multiple Range Test for a maximum of 15 correlated and heteroscedastic means. It has an option to input either the variance-covariance matrix of the estimates of the corresponding treatment means, or the number of replicates in each mean. The procedure used is described in D. B. Duncan's paper "Multiple Range Tests for Correleted and Heteroscedastic Means", Biometrics, 1957, Vol. 13, pages 164-168. The complete package consists of the mainline program, three subroutines and short supplementary program which stores the tables of 5% and 1% Significant Studentized Ranges, required by the main program, on disk.

PROGRAMMING SYSTEMS - Written in FORTRAN IV and has been compiled and tested on an 1130 Model 2B using Disk Monitor system Version 1, Model 5.

MINIMUM SYSTEM REQUIREMENTS - An 1131 CPU Model 2B with 8K words of core, 1 disk drive, an 1132 Printer and a 1442 CRP.

BASIC PROGRAM PACKAGE
DOCUMENTATION - Write-up.
MACHINE READABLE - Source code and sample problem.

OPTIONAL PROGRAM PACKAGE - None.

ORDERING INFORMATION: PROGRAM NUMBER 1130137001

	PROGRAM NUMBER EXTENSION	DISTRIBUTION MEDIUM TYPE	MEDIUM CODE	USER VOLUME REQUIREMENT
BASIC	none	CARDS	15	none
OPTIONAL	none	none		none

* *

ACCUMULATIVE LIST OF NEW CONTRIBUTED PROGRAMS

Following is an accummulative list of new Type III and Type IV programs which have been made available since the last edition of the Catalogs of Programs or their Supplements. Abstracts describing these programs can be found in the Installation Newsletter Issue No. shown below.

Program Number	Program Title	Newsletter Issue No.
360D-00.1.021	S/360 GENERALIZED EXTRACTION AND MATCH PROGRAM	69-07
360D-00.1.022	NYU PRINT-UTILITY PROGRAM	69-07
360D-00.2.006	OS SYSTEM/360 FLOWCHART (OS FLOW): DOS FLOWCHART UNDER OS/360	69-03
360D-00.3.025	S/360 DOS TAPE TO PRINTER/PUNCH SPOOLING SYSTEM	69-08
360D-00.5.005	FORMATTED LISTING OF VTOC FOR OS/360	69-08
360D-01.1.006	COMPILE AND GO FOR THE OS 360, PL/I (F) COMPILER	69-06
360D-03.0.007	S/360 MOD 44 MULTIPROGRAMMING SYSTEM	69-08
360D-03.0.008	DOS IOCS LOGIC MODULE SOURCE STATEMENT GENERATOR	69-08
360D-03.2.011	OS/360 PL360 COMPILER	69-06
360D-03.3.008	COURSE III/OS A REAL-TIME COMPUTER ASSISTED INSTRUCTIONAL SYSTEM	69-01
360D-03.4.026	BSAM SUBROUTINE PACKAGE FOR OS 360, VERSION I	69-06
360D-03.4.027	FORTRAN RANDOM I/O SUBROUTINE	69-03
360D-03.4.028	2260 LOCAL TERMINAL USERS SYSTEM (LOTUS)	69-06

360D-03.6.010	JOB STREAM MANAGER OS/MVT	69-03
360D-03.7.025	CARRIER RETURN/LINE FEED ON THE 1052/53, MISLCRLF	69-03
360D-03.7.027	EDIT MACRO FOR DOS, TOS, OR OS ASSEMBLY LANGUAGE, S/360	69-07
360D-03.8.013	PL/I STRING FUNCTIONS	69-06
360D-04.0.008	DOS COBOL CROSS REFERENCER	69-04
360D-04.1.010	DEBUGGING AID FOR DOS PROGRAM TESTING	69-04
360D-04.2.007	PROGRAM TO AID IN OPTIMIZING COMPUTE BOUND JOBS BY INDICATING THE MAJOR PROCESSING LOOPS	69-04
360D-04.6.001	A PROGRAM FOR GRAPHIC ON-LINE DEBUGGING ON AN IBM 2250 DISPLAY	69-08
360D-05.2.010	INTERACTIVE APPLICATIONS SUPERVISOR (IAS/360)	69-06
360D-06.3.006	GENERALIZED TELEPROCESSING PROGRAM FOR SYSTEM/360	69-07
360D-06.3.007	DOS BTAM FOR LEASED LINE KSR 35 TELETYPES	69-06
360D-06.5.006	UNIVAC-1108 TO IBM-360 FLOATING POINT INTERNAL CONVERTER ('CVFL08')	69-06
360D-06.5.007	7090 TO 360 FLOATING POINT INTERNAL CONVERTER ('CVFL09')	69-06
360D-06.6.003	FORTRAN CHARACTER STRING PACKAGE	69-03
360D-06.6.004	CHARACTER FILTER PL1	69-03
360D-06.7.018	BSEARCH - A RANDOM ACCESS BINARY-SEARCH TECHNIQUE FOR SEQUENTIAL FILES ON DISK OR DRUM	69-03
360D-06.7.019	KWADE-KEYWORD AS A DICTIONARY ENTRY	69-03
360D-07.6.001	2741 READ AND WRITE ROUTINE	69-06
360D-10.0.001	PROGRAM TO CALCULATE THE AMOUNT OF CPU TIME AVAILABLE IN THE BACKGROUND WHEN RUNNING DOS MULTIPROGRAMMING	69-03
360D-10.0.00	FORTRAN ANALYSIS PROGRAM	69-04
360D-13.1.005	GENERAL TAXONOMY PROGRAM	69-08
360D-13.4.007	RESCALED RANGE TRANSFORM	69-06
360D-16.2.018	BULKHEAD SHEET PILING ANALYSIS	69-06
360D-17.7.001	SUBROUTINE UCONIC	69-07
360D-19.7.010	PROOF OF DEPOSIT ENTRY RUN WITH TRANSIT DISTRIBUTION	69-08
360D-19.7.011	PERSONAL TRUST ACCOUNTING PACKAGE TO ASSIST USERS IN INSTALLING OFF-LINE PROCESSING	69-07
360D-25.0.004	EQUIPMENT ORDER, SHIPMENT, INSTALLATION, AND DISCONTINUANCE REPORT FOR LARGE ACCOUNTS	69-08
360D-25.2.004	UNIT CONTROL SYSTEM FOR RETAILERS	69-08
360D-42.2.001	EXPERIMENTAL PROGRAM FOR DETERMINING POLYNOMIAL ZEROS	69-06
360D-45.1.001	INVRSOL	69-03
360D-45.2.001	EIGENRS	69-03
360E-01.0.001	INVALID CCCP/360-20 INVALID CODE CARD CORRECTION PROGRAM	69-07
360E-01.0.002	FIELD SELECTION AND LIST	69-08
360E-01.5.001	MODEL 20 STAND-ALONE DISK COPY UTILITY	69-07
360E-03.4.002	SHORT INPUT OUTPUT SYSTEM	69-06
360E-06.5.002	JULIAN DATE PROCESSING SUBROUTINES	69-08
360E-13.0.001	HOSPITAL PBAR PROGRAMS FOR A MFCM MODEL 20	69-07
360E-13.0.002	STATISTICAL COMPARISONS PROGRAM	69-07
360E-13.6.001	LINEAR REGRESSION IN TWO VARIABLES (LEAST SQUARE FIT) FOR THE MOD 20	69-06
360E-19.7.020	SAVINGS ACCOUNTING FOR THE 360 MOD 20 DISK	69-04
360E-28.3.014	WHOLESALE AND/OR RETAIL, ACCOUNTS RECEIVABLE, AGED TRIAL BALANCE	69-06
1130-00.3.010	1130 PRNZ MODIFIED FOR ADDITIONAL CARRIAGE CONTROL	69-07
1130-06.3.001	IBM 1130 REMOTE JOB ENTRY SYSTEM	69-06
1130-06.6.014	THE 1130/1800 KEYWORD INDEX GENERATOR	69-04
1130-06.6.015	IBM 1130 FORTRAN REREAD DEVICE AND FREE FORMAT KEYWORD INPUT ROUTINES	69-06
1130-13.7.001	MULTIPLE RANGE TEST FOR CORRELATED AND	

	HETEROSCEDASTIC MEANS	69-08
1130-17.3.002	GENERAL PURPOSE CHROMATOGRAPH PEAK INTEGRATION PROGRAM	69-07
1130-28.2.002	1130 PATIENT BILLING	69-08
1130-28.3.001	1130 HOSPITAL ACCOUNTS RECEIVABLE	69-08
1130-30.1.004	INDIVIDUALIZED HOMEWORK ASSIGNMENT AND QUIZ GENERATOR (QUIZ AND EXERCISE DEMONSTRATOR: Q.E.D.)	69-04
1800-00.0.002	OBJECT TIME I/O LOGICAL NUMBER GENERATOR	69-04
1800-23.5.006	IBM ANALOG DATA ACQUISITION PROGRAM	69-04

IBM Installation Newsletter

September 26, 1969 Issue No. 69-19

CONTENTS

(G means Green Section)

Distribution: Branch Office-DP Management, Salesmen, Systems Engineers, FE Managers. Regions, Districts, Education Centers, Field Systems Centers, Federal Systems Centers, FE Area Offices, DPD HQ, FED HQ, WTC.

*Requires Immediate Attention

SELECTED SHORT SUBJECTS

The purpose of Selected Short Subjects is to bring together and highlight concise, factual and timely information which will indicate that action is to be taken by the IBM representative whose accounts are affected.

1. 1600 BPI PE Tape Drive Manual

A new IBM Confidential Sales and Systems Guide is now available from Mechanicsburg entitled "1600 BPI Phase Encoding and the IBM 2420 Tape Drives". The form number is Z20-1948.

This marketing guide is intended for Marketing Representatives and Systems Engineers. It discusses the significance of 1600-bpi phase encoding and 2420 Magnetic Tape Units as they relate to system performance. Conversion techniques and justification for converting to these tape drives are discussed. This guide also includes references to recent successful conversions to 2420 Magnetic Tape Units.

OS/360 RELEASE 18 PLANNING INFORMATION

As indicated in the Sales Manual, the next OS/360 release (#18) is planned for October 31, 1969 availability from PID. Release 18 is projected to include those items committed for October 31, maintenance and a number of incremental improvements. An index of items listed in this article is as follows:

- Components Announced for October 31, 1969
- Control Program Incremental Improvements
- Data Management, IOS and Device Capability Incremental Improvements
- SYSGEN Incremental Improvements
- Serviceability and Reliability Incremental Improvements
- Utilities - Incremental Improvements
- Graphics
- Compiler and Language Support Incremental Improvements
- Component Releases and Updated Systems
- Component Support on Prior Systems
- PTF and Emergency Bypass Support

Components Announced for October 31 Availability

System Management Facilities (SMF) (P69-62):
The SMF option available to MVT users gathers basic system information and statistics about each job's use of the CPU, core storage and I/O devices. Also included is support for GJP improved from a June 30, 1970 availability date.

USA Standard COBOL (P68-145 and P69-88):
A new COBOL compiler supporting the USA Standard COBOL will be an integrated component of Release 18. This new compiler which replaces COBOL F retains COBOL F's advanced features and includes new features of the USA Standard COBOL. COBOL F will continue to be distributed and maintained for two years following the initial availability of USA COBOL in Release 18. The Language Conversion Program (LCP) will not be part of Release 18, but will be made available as a separately orderable item.

ISAM - Variable Length Records (VLR) (P69-97):
The Indexed Sequential Access Method will provide the ability to create and maintain Variable Length Records. In addition, the data set utility IEBISAM will handle Variable Length Records.

Model 85 Support (P68-44):
Programming Support for the Model 85 under MVT or MFT will be integrated into Release 18. The Model 85 has new features which require programming system support extension beyond that now in OS/360. These extensions, which are in addition to the MVT and MFT capabilities already announced, include support of:

- CRT Operator Console
- Extended precision floating point
- Recovery Management Support
- Assembler F Extensions

Multiple Console Support (MCS) (P69-30):
Multiple Console Support is an option provided under MFT, MVT and M65 MP. It enables a Multiple Console configuration, which provides message routing to functional areas and increased systems availability through alternate consoles and console switching.

Model 65 Multiprocessing 2K Storage Reconfiguration (P69-14):
Storage reconfiguration provides high system availability by permitting deallocation of operating system resources from an interrupted job, and by performing logical isolation of failing storage in 2K byte elements.

BTAM Support for 2760 Optical Unit (P69-52) and 2741 Communication Terminal (P69-23):
BTAM support for the 2760 and the 2741 will be provided with Release 18.

Model 65 MP Graphics (P69-39):
Support listed below will now be available:

1. Graphic Programming Services
2. Graphic Programming Services for FORTRAN
3. Graphic Job Processor
4. FORTRAN Subroutines for Data Transmission between a S/360 M65 MP and an 1130
5. Job Control from an 1130 to a 2250 using the Satellite Graphic Job Processor.
6. Asymmetric support for the 1443, 2150, 2501, 2250, 2260, 2701, 2702, 2703, 2840 and 2848.

Unannounced Incremental Improvements Planned for Inclusion in Release 18

Control Program:

. Resident Link Library Option in MFT-Provides an option for making re-entrant link library modules resident in main storage for MFT users.

. Dedicated Workfiles in an MVT Environment-Allows for pre-allocation of workfiles as specified in cataloged procedures for initiators. The allocation and disposition takes place when the initiator is started and stopped. This eliminates envoking the DADSM allocate and scratch routines for each job step requesting workfiles.

. Reduction of Device Mask and Name Tables - Specific unit entries have been eliminated from both the device mask table and device main table, thus, reducing the size of these tables. The amount of core saved per device increases with the number of devices sysgened. For example; there will be approximately a 2K savings when 60 devices are sysgened.

. Step Dispatching Priority-Allows the user to more effectively utilize system resources by specifying dispatching priorities for each step or selected steps of a job.

. ABTERM/ABEND-Remove from ABTERM/ABEND the ability to place a system in a wait-state. A damage assessment routine will process those conditions that previously led to a wait-state in an effort to maintain task and system operation without interruption.

. Retained Volume Improvement - Allows the user to more effectively control volume traffic in RETAINed volumes and volumes containing PASSed data sets.

. K Size Memory for Model 65 Multiprocessing-Allows for up to 4 additional 2365 - 13 core storage units to be attached to the model 65 multiprocessor.

. 2870 Selector Subchannel Separation - Device Allocation has been modified to treat selector subchannels on a 2870 channel as separate paths for data.

. Start Command Symbolic Parameters-The start facility has been improved to permit symbolic parameter substitution and to permit override of any statement of a catalog procedure for a STARTed task.

. MSGLEVEL Improvement - This facility has been modified to permit the user to control the categories of messages he desires to have issued.

. Universal Channel Error Support - Provides one set of device dependent error recovery procedures which contain full channel error support for CPU models 65 and higher.

. Mod 65 Recovery Management Support - Extended to include the multiple console support option.

Data Management, IOS and Increased Device Capabilities

. Expanded Device Support-Increases to 768 the number of I/O devices that can be supported. It also supports a second 2870 multiplexer channel.

. IOS Ordered Seeked Queueing-Minimizes arm movement for seeks to direct access devices.

. DOS to OS VTOC Conversion-Enables the use of standard OS/360 options for DASD allocation on volumes which have DOS data sets. This does not convert or permit reading of DOS data under OS, but enables packs initialized as DOS Packs to be used with OS.

. User Totalling Facility-Provides the user with the ability to maintain a control total as he creates a data set and to have that total synchronized with the records actually written on each volume using SAM.

. User Labels for Direct Access Devices-Now supported for data sets on direct access devices.

. MOUNT/DISMOUNT Message Improvements-Provides additional information to the user

about the volumes being mounted or removed.

. Re-write Volume Label-The use of scratch output tapes has been improved by causing OPEN and EOV to re-write all standard volume labels, to create or destroy labels as specified by the user, and ask for a new volume instead of giving an ABEND.

. BTAM-Manual dial/answer WTC modems are supported. Also, a number of revisions to BTAM binary synchronous communication support has been made to enhance the reliability and compatibility characteristics of the system.

. 2420 Model 5 Tape Unit-Support now provided.

. RJE Improvement Package-Provides increased throughput on the 2780 and CPU work stations operating on medium speed lines, reduces the amount of core storage needed for the RJE region or partition and improves the maintainability of the RJE system.

SYSGEN

. I/O Device SYSGEN-Allows the user to add devices to an existing system without requiring a complete operating system generation.

. Multiple Address Generation-Allows the user to specify a group of like devices on one IODEVICE card.

. User Job Card for Stage Two of SYSGEN-Allows the user to specify his own job card for stage II during stage I.

Serviceability and Reliability

. System Environment Recording (SER)-Routines have been provided to support the Model 40-192K and the Model 50-384K storage sizes.

. OLTEP-Modified to allow the running of tape gap test on a tape drive attached to the hi-speed multiplexer channel O.

. Utilities

. IEHLIST improvement - The LISTVTOC FORMAT option has been expanded to include additional information

. SCRATCH VTOC SYS-Function of IEHPROGM extended to scratch MVT created temporary data sets.

. Data Set Generator-Expanded to create and access records of an indexed sequential data set. Also, to select fields from an input record to be used in creating an output record.

. Control Card Scan-Improved diagnostics provided.

Graphics

. GSP In Core Data Conversion Routines-Provides the FORTRAN programmer with the ability to dynamically change data formats.

. GSP Performance Improvement-Improved coding eliminates loading of more than one copy of a GSP routine having an alternate entry point.

. GSP-SYSOUT Data Set Display-Provides flexible routine enabling the user to decide if he wants hard copy of data sets created by GSP foreground jobs after viewing the data sets.

Compilers and Language Support

. Revised FORTRAN Library Math Functions-The FORTRAN Library has been updated to take advantage of the improved floating point hardware. Certain algorithms have also been revised.

. FORTRAN VRE Support-Has been brought up to the standards for the spanned record type and will thus be compatible with data management creation and processing of such records. Previously created FORTRAN data sets are still usable within FORTRAN.

. Assembler F OS/DOS Parm Field Option-Causes Assembler F to perform as DOS Assembler F. It will flag Q-CONS, DXD's, and CXD's and will not sort the RLD entries.

. COBOL E NO SPIE Option-Provides the programmer the ability to surpress the generation of SVC 14 and provide his own assembly language program interrupt analysis routine.

. RPG-Externally conditioned indicators can be specified to control the execution of the program or functions within the program. In addition, RPG will allow the creation of and additions to an indexed sequential data set.

. Linkage Editor E-The diagnostic messages for synchronous errors are expanded to give more detailed information about the failing I/O operation.

. Compilers Statistics-Various compilers will provide summary information about each source program compiled. This will include options in effect, the number of source statements, and program size.

Component Releases and Updated Systems

It is our intent to provide component releases for use with Release 18 after the initial availability of Release 18. A planned list of component releases will be provided with the Release 18 Announcement Letter.

It is also our intent to provide an updated Release 18 similar to the updated Release 17 previously provided. This updated Release can contain PTF's, maintenance and release independent components. The plan is to make the updated release available 3 to 4 months following the initial availability of Release 18. Any component releases made available prior to the updated release will be a part of the updated Release 18. Those users not requiring the new function or maintenance in Release 18 upon its initial availability will have the opportunity to defer ordering and installing Release 18 until the updated version becomes available.

Component Support on Prior System Releases

Many components such as compilers, sorts and utilities are release independent and may be used with and will be supported on a prior release. Details on the Release 18 components that will be supported on Release 17 will be provided in the Release 18 announcement letter and memo to users. This will enable the Release 17 user to obtain the maintenance in these components, thus, providing him with an improved system until he decides to install Release 18 or the updated Release 18.

PTF and Emergency Bypass Support (Sales Manual GI 65.6)

FE will apply PTFs or attempt to develop emergency bypasses when required on current systems. An OS release is "current" for 9 months after the availability of the next release from PID. Release 15/16 is "current" and will be supported with PTFs through December 24, 1969. Release 17 will continue to be a current release for 9 months after the availability of Release 18. Thus, the user has a number of options; install selected components of Release 18 on his "current" system, install Release 18 or wait for the updated Release 18.

OS/360 ADDITIONAL DIDOCS PLANNING INFORMATION

This article provides additional information about the Device Independent Display Operator Console Support (DIDOCS) announced on August 20, 1969 for delivery by September 30, 1970. The support includes all S/360 configurations that support OS/360 MFT, MVT, and MP65, in addition to the Model 195 which is supported for MVT only.

The significant points to multiprogramming users other than the Model 195 are:

- The 2260 is supported as an operator console under Multiple Console Support (MCS). It can be used as a primary console or a remote console.

- The 2250 Models 1 and 3 are supported as an operator's console under MCS for MFT, MVT and MP65. 2250 Model 1 was previously supported only for MVT users.

- The current 2250 Model 1 support will be replaced when DIDOCS becomes available. Models 50, 65, 67 in 65 mode, 75 and 91 users must add Multiple Console Support to use the 2250 as a console. Model 85 has always required MCS. The usual extended support policy for maintenance of the replaced 2250 support applies.

The 2260 connected to a 2848 Model 3 by the Direct S/360 Channel Attachment with Alphanumeric Keyboard and Line Addressing will be supported as a console device similar to the 2250 and Model 85 Operator Console Feature Code 5450. The 2260 has a 12-line screen of 80 characters per line. The top 8 lines are devoted to system messages, and the bottom 4 are used as a system control area. New operational improvements announced for the 2250 Console will be included. Highlighting of action messages, Automatic Message Deletion Mode, and Roll Mode are examples.

The 2260 non-destructive cursor is supported as an optional feature. This feature is recommended as it permits the operator to backspace and correct errors. The operator does not have to retype the entire error line. He can change only the characters in error if desired. The non-destructive cursor can be used to delete action messages directly. Experience with the 2250 shows these features to be of significant advantage.

The Multiple Console Support required for the DIDOCS and the 2260 includes a message routing feature. This can be used effectively to limit the amount of message traffic to a given operator station. The 2260 with its 8 lines of system messages can be significantly more effective than a device such as a typewriter or printer. This is, in part, due to the full view of all appropriate messages in an upright position. There is a special deletion method for action messages as they are responded to by the operator. Other human factors such as quietness of the display unit reduce operator fatigue.

DIDOCS is consistent with the announced Model 85 Operator Console Support and takes advantage of device dependent features of each display such as the light pen on the 2250 for message deletion. The support will provide uniform displays within the constraints of each of the three device types supported.

Core Storage Planning Information

For MCS

Basic Multiple Console Support	
resident module for MFT	1.8K
resident module for MVT	2.6K
Hardcopy Log on System Log for MFT (No change for MVT) BSAM must be resident	3.0K
User exit routines will take additional core as needed	----
M85 Operator Console Feature 5450	4.0K
2740 Console without BTAM present in the system	6.2K
plus 2.6 bytes for the first 2740	2.6K
2740 Console with BTAM present in the system	none
plus 2.6K bytes for the first 2740	
Each console SYSGENed	0.3K

For DIDOCS

Each Display Type Console SYSGENed	200 Bytes
2250 on Model 91/195 Display Console	5K
2260 Display Console	2K
Model 85 Operator Console	4K

DOS/360 TIMESAVER GUIDE

- Unexpected DOS problems and no solution?
- Can't figure what the SRL means?
- Need a concise "here's-the-problem-and-solution" manual?

The DATACENTER DOS TIMESAVER is a "cookbook" guide for preventing or solving time-consuming DOS problems.

Covers:

JCL examples
utility customizing
core print-out
dump-restore (DOS and OS)
DEBE
DITTO
operational problems
MPS operational problems
DOS FE tips

Available from Mechanicsburg as a Tie paper. (Form number Z77-9191-0.)

INFORMATION ABOUT THE NEWSLETTER

EVERY SALESMAN AND SYSTEMS ENGINEER SHOULD RECEIVE ONE COPY OF THE NEWSLETTER. Each FE Branch Manager should receive five copies for distribution to Customer Engineers.

The IBM Installation Newsletter is distributed only to IBM locations and is not available to customers. The Newsletter is separated into two parts. The first part contains information which is intended only for IBM personnel and the pages are labeled For IBM Internal Use Only. THESE PAGES, OR REPRODUCTIONS OF THEM, ARE NOT TO BE GIVEN TO CUSTOMERS. However, using your own good judgement, you may discuss pertinent information from these pages with a customer. The second part contains information which may be REPRODUCED by the Branch Office at their discretion and given to customers. These pages are not labeled with any restrictive classification. Note that when any abstracts for Type III programs are given to the customer, he must be informed that the disclaimer for Type III programs as contained in the Newsletter, applies. YOUR COOPERATION IN PROPERLY HANDLING THE CONTENT OF THE IBM INSTALLATION NEWSLETTER IS NECESSARY FOR ITS CONTINUED PUBLICATION.

Changes to the quantity of the Newsletter shipped are to be made by the Administration Manager. (NOTE THAT EXTRA COPIES OF THE NEWSLETTER ARE NOT TO BE ORDERED FOR DISTRIBUTION OF INFORMATION TO CUSTOMERS.) He should update and send the DP Administration Managers' Document Control card (Electro IBM N61508) to DP Document Control, DPD HQ.

Direct inquiries concerning the contents of this Newsletter to:

W. F. Malmborg, Editor
IBM Installation Newsletter
Dept. 812, DPD HQ
112 East Post Road
White Plains, New York 10601
Telephone 914 Wh9-1900 Extension 6158
Tie Line 374-6158

NEW PUBLICATIONS ANNOUNCED BY PRL'S

Information is extracted and condensed from the weekly PRL's (Publications Release Letters) to insure that all Salesmen and Systems Engineers are aware of new or revised Reference Sources, Marketing Publications, and Field Engineering Publications. Normally, each issue of the Newsletter will contain information extracted and condensed from two PRL's, one following the other. The information will be placed in the Newsletter in its original sequence with no rearrangement of form numbers or titles. It is not intended to replace existing information and distribution sources. You should be certain that you are aware of these sources.

Do not use the condensed Newsletter information to order publications. Most but not all (for example, some films and slide sets) of the items are available from Mechanicsburg. To receive the items you need without delay, prior to ordering check the REQ (requisition Form) code in the "Accumulative Supplement to IBM Publications Current Price List (Z20-0100)" which is attached to the weekly Publications Release Letter (PRL) and distributed to your Administration Manager. For an explanation of the REQ codes and other publications information see the first few pages of the "IBM Publications Current Price List Z20-0100."

The IBM publications requisition (MO2-0618) or ITPS should be used for ordering items from the IBM Distribution Center, Mechanicsburg, Pa.

(IBM World Trade Corporation has its own distribution center and procedures.)

MARKETING PUBLICATIONS

PRL #36 September 5, 1969

FORM NO.	TITLE	DISPOSITION
520-2175-0	DACS Centers	NEW
A10-7000-0	IBM System/3 Installation Manual-Physical Planning (Spanish Translation)	NEW
C10-7001-0	IBM System/3 Card System RPG II and Sort/Collate Reference Manual (Spanish Translation)	NEW
C10-7002-0	IBM System/3 Card System Introduction (Spanish Translation)	NEW
C10-7003-0	IBM System/3 Data Recording and Data Verifying Programs Guide (Spanish Translation)	NEW
C10-7006-0	IBM System/3 Disk System Introduction (Spanish Translation)	NEW
C26-3718-3	IBM 1800 Multiprogramming Executive Operating System Introduction	SCRAP-C26-3718-2 N33-8027-0
C30-1005-0	IBM S/360 OS/DOS Planning for Improved BTAM Support of Remote BSC Stations	NEW
L27-3025-0	IBM 2740-i/2950-2 Batch Buffer Terminal Installation Manual -- Physical Planning	NEW
N20-2039-0	TNL to System/360 Continuous System Modeling Program (360A-CX-16X) User's Manual Re: H20-0367-2	NEW
N22-0342-0	IBM System/360 Installation Manual Physical Planning	NEW
N26-0246-0	IBM System/360 Component Descriptions Re: A26-3599-3	NEW
N33-8609-0	S/360 BOS Operator Messages Form C24-5024-3	NEW
R20-6005-0	System/3 Application Design Case Study Exercises	NEW
R20-9304-0	IBM Basic Punched Card Procedures	NEW
R29-0206-4	System/360 COBOL-COBOL Program Fundamentals	UES-R29-0206-3
R29-0215-3	IBM System/360 COBOL - COBOL Programming Techniques	UES-R29-0215-2
R29-0249-2	IBM Computing Systems Fundamentals	UES-R29-0249-1

FIELD ENGINEERING PUBLICATIONS

123-0453-3	2020-II Processing Unit	SCRAP- S23-9047 S23-9051 123-0453-2
123-3104-0	2914-1 Switching Unit	NEW
229-0029-0	Installation Planning Checklist	NEW
R23-3060-2	Introduction to Computers - Student Self-Study Course	UES-R23-3060-01
R31-0423-0	1287 Optical Rd. Mod 3 & 4 - Student Guide	NEW
S27-0655-0	IBM 1800 Processor - Controller (123-0483-2)	NEW
S32-0002-0	FES 2401/2402 - Model 5 & 6	NEW
Y25-2723-1	2701 Data Adapter Unit	SCRAP - Y25-2723-0

REFERENCE SOURCES

PRL #37 September 12, 1969

FORM NO.	TITLE	DISPOSITION
N20-1002-26	DPT Newsletter Re: F20-8172-6	Scrap N20-1002-25

MARKETING PUBLICATIONS

270-0028-3	9950/54 Micro Copier - IPC	Scrap 270-0028-2
270-0053-2	9951/56 Physical Planning Manual	Scrap 270-0053-1
270-0103-0	9965 Micro Copier - IPC	NEW
570-0449-0	Federal Supply Schedule Tabulating Cards	NEW
C24-9001-5	S/360 Model 20 DPS/TPS RPG SRL	Scrap C24-9001-4 & N33-9052
C33-4001-4	S/360 Model 20 CPS/DPS/TPS BSCA IOCS	Scrap C24-4001-3
C33-4003-1	S/360 Model 20 Remote Job Entry Work Station	Scrap C33-4003-0
H20-0599-2	S/360 Retail IMPACT Staple System (DOS) Version 2 Program Description Manual	UES H20-0599-1
H20-0635-0	Information Management System/360 for the IBM S/360 - Operations Manual Volume 1 - Systems Operation - Program Number 360A-CX-31X	NEW
H20-0634-0	Information Management System/360 for the IBM S/360 Program Description Manual	NEW
H20-0636-0	Information Management System/360 for the IBM S/360 Operations Manual- Volume 2 - Machine Operations	NEW
K20-0328-0	Commercial Loan Accounting Using the IBM S/360 with 2260 Display Stations	NEW
L22-6937-0	IBM S/360 Custom Feature Description - 2914 Switching Unit Model 1 - RPQ 880882	NEW
N20-2012-0	TNL to Information Management System/360 for the IBM System/360 ADM Re:H20-0524-0 & -1	NEW
N27-3042-0	TNL to IBM 2701 Adapter Unit Component Description Re: A22-6864-3	NEW
N27-3044-0	TNL to IBM 2702/2703 Transmission Control Original Equipment Manufacturer's Information Re: A27-3012-0	NEW
N33-1528-0	TNL to IBM S/360 Model 20 Functional Characteristics Re: A26-5847-4 & -5	NEW
N33-8066-0	IBM S/360 Model 20 Remote Job Entry Work Station Re: C33-4003-1	NEW
N33-8600-0	TNL to S/360 Model 20 TPS Operating Procedures Re: C24-9009-3	NEW
N33-9062-0	TNL to S/360 Model 20 DPS Control and Service Programs Re:C24-9006-4	NEW

PRL #37 (Continued)

N33-9063-0	TNL to S/360 Model 20 Operating Procedures Re: C33-6004-3	NEW
N33-9064-0	TNL to S/360 Model 20 DPS Performance Estimates Re: C33-6003-2	NEW
N33-9065-0	TNL to S/360 Model 20 Disk Sort/Merge Re: C26-3806-4	NEW
R20-0065-1	Property and Liability Information System - Education Material - Volume 4 Record Formats - Homeowners	Scrap R20-0065-0
R20-9151-4	IBM S/360 COBOL - Writing Programs in COBOL	Scrap R20-9151-3
R20-9152-4	IBM S/360 COBOL Techniques - COBOL Programming - PI	Scrap R20-9152-3
R20-9292-1	6430 Programming Course Description	Scrap R20-9292-0
R29-0145-2	Punched Card Data Processing Principles - 85 Collator Operation and Wiring Text	UES R29-0145-1
R29-0231-4	S/360 Assembler Language Coding System Review Text	UES R29-0231-3
R29-0252-1	S/360 Model 20 Assembler Language Coding Instructions	UES R29-0252-0
V25-6199-1	IBM Magnetic Board Pieces	UES V25-6199-0
X26-5961-6	IBM S/360 Model 20 Physical Planning Template	Scrap X26-5961-5
Y20-0294-1	Mathematical Programming System/360 Report Generator System Manual (360A-CO-20X)	UES Y20-0294-0
Y20-0396-0	1130 Mechanism Design System-Kinematics System Manual, Volume 1 - Flowcharts Narratives Program Number 1130-EM-03X Program Application Bulletim	NEW
Y20-0453-0	TNL to CALL/360 Internal Specifications Manual, Volume 6 Re: Y20-0388-0	NEW
Y28-2407-0	TNL to S/360 Operating Systems Consolidated Document Re: Y28-6681-1	NEW
Y28-2408-0	TNL to S/360 OS Colsolidated Document Re: Y28-6681-1	NEW
Y33-8004-1	S/360 Model 20 IOCS for the Binary Synchronous Communications Adapter PLM	Scrap Y33-8004-0 & TNL's Y33-8014 & Y33-9037
Y33-8006-0	S/360 Model 20 Remote Job Entry Work Station Program Logic Manual	NEW
Z23-6009-0	System/3 Installation Control For Management - Examination	NEW
Z29-1954-1	Manager's Review Guide - Branch Marketing Assessment Program	Scrap Z29-1954-0
Z29-1996-1	Manager's Review Guide - Basic Systems	Scrap Z29-1996-0
Z77-8013-1	Reconstruction of a Nicked or Damaged Strip in the Middle of any Index Sequential File on the Data Cell	UES Z77-8013-0
Z77-9186-0	Job Accounting, Benchmark Selection and Cost Performance Preduction for Multi-programming systems	NEW
Z77-9187-0	Development of a Large Hospital Data Bank	NEW
Z77-9189-0	Commercial OS - Modifications and Additions	NEW
Z77-9191-0	Data Center DOS Timesaver	NEW

FIELD ENGINEERING PUBLICATIONS

123-0414-3	2403/2404 Model 2 & 3 Magnetic Tape Unit & Control Illustrated Parts Catalog	Scrap 123-0414-2
123-0461-2	2314 Direct Access Storage Facility Illustrated Parts Catalog	Scrap 123-0461-1
229-2193-0	IR Code Guide for S/360 Model 25 CPU - FE Reference Card	NEW
R23-3118-0	Channel Adapter Student Self Study	NEW
R25-5364-2	2703 Student Guide	UES R25-5364-1
R27-9598-0	2420 Model 7 Instructional Diagrams	NEW
R31-0291-3	Checkpoint A Review Sheet	Scrap R31-0291-2
R31-0292-3	Check Point B, Review Sheet Course Development Worksheet	Scrap R31-0292-2
R31-0293-3	Checkpoing C Review Sheet Course Development Worksheet	Scrap R31-0293-2
R31-0294-2	Final Approval Review Sheet Course Development Worksheet	Scrap R31-0294-1
R31-0458-0	System/3 Architecture Supplementary Course Material	NEW
S29-3556-0	Supplement to R31-0299-0	NEW
S29-3558-0	Supplement to R31-0296-1	NEW
S29-3559-0	Supplement to R31-0095-4	NEW
V31-0123-0	L287/L288 Optical Reader Preschool Microfiche Card	NEW
V31-0178-0	System/3 Architecture Microfiche Card	NEW
Y22-9506-5	2401/2/3/4 and 2803/4 FE Installation Manual	Scrap Y22-9506-4
Z25-2173-1	Instruction for Obtaining Reimbursement - Student Instruction Book	UES Z25-2173-0
Z25-3526-0	FE & WT Course Material Listing	NEW
Z31-0202-0	5410 Central Processing Unit FE Maintenance Diagrams	NEW
Z31-0207-0	5410 Central Processing Unit FETOM	NEW
Z31-0221-0	Off-Line Keypunch FEMDM - Preliminary	NEW
Z31-0230-0	5424 Multi-Function Card Unit FEMM - Preliminary	NEW
Z31-0240-0	5203 Printer Attachment FEMDM - Preliminary	NEW
Z31-0244-0	5410 Central Processing Unit FEMM - Preliminary	NEW
Z31-0245-0	5203 Printer Attachment FETOM- Preliminary	NEW
Z31-0253-0	5424 Multi-Function Card Machine Attachment FETOM - Preliminary	NEW

September 26, 1969 **IBM INSTALLATION NEWSLETTER** Issue No. 69-19

Table of Contents

September 26, 1969 **IBM INSTALLATION NEWSLETTER** Issue No. 69-19

OS/360 MVT REGION DETERMINATION CLARIFICATION

The following is a clarification of the procedure followed to determine the region size used to initiate, execute, and terminate a job step.

When a job is first interpreted, a region request is needed for the Step Control Table. If one is not provided on the JOB or EXEC statement, the default is not to MINPART, but to the size specified in the parameter field of the reader procedure. (See System Programmer's Guide, C28-6550.) For example, the default size provided by a reader procedure at one installation was observed to be 100K whereas the procedure distributed by DPPID has a default of 50K.

If the initiator/terminator (IEFSD061) and allocation (IEFSD062) modules are in the Link Pack area, the following happens: if the region request, either explicit or default, is less than MINPART, the latter will be used to initiate the step and then the region will be freed and the larger of the region request or the termination requirement will be obtained to execute and terminate the step; otherwise the region request will be used to initiate, execute, and terminate the step. (See MVT Job Management PLM, Y28-6660.)

The above means that if MINPART is 52K and the reader procedure provides a default of 50K, a step which uses the default region size will introduce an overhead of releasing a region of 52K to obtain 50K.

S/360 Model 25-1401 Compatibility Mode and Executing a FARGO One-Six Sixty-G Program

If the user has implemented the "retest" function of Fargo, it is possible to lose part of a print line because of overflow. The sequence of instructions in the control portion of Fargo is as follows:

TEST 1	B HEAD, @	branch to heading on channel 12
	NOP X, X	NOP
	B TEST 1 P	branch if printer is busy
	B Process	process detail record

Due to the internal interpretive routines in the model 25, it is possible to execute the first instruction while the printer is still busy and not yet at channel 12, have the print complete and reach channel 12 during the interpretation of the NOP instruction, and hence fall through the next test.

Then if part of a print line were set up and the retest occurred before printing took place (the next time through the above loop), the first branch would be taken and that portion of the print line that was set up would be wiped out by the heading routine.

To correct the above problem, the author used 3 patch cards. Those cards will work either in the Fargo compiler or in the condensed object decks. It is not necessary to recompile any Fargo Programs. Merely place the patch cards immediately ahead of the last card of the object deck.

S/360 MODEL 25-1401 COMPATIBILITY MODE AND EXECUTING A FARGO ONE-SIXTY-G PROGRAM

IBM GENERAL PURPOSE CARD PUNCHING FORM — X20-8030-1 Printed in U.S.A.

JOB FARGO 160-G PATCH
BY
DATE 9/1/69

PUNCHING INSTRUCTIONS: WRITTEN AS: / PUNCH AS:

NOTES:

FIELD IDENTIFICATION

Line	Columns 1-10	Columns 41-80
1	B647E	L005980, 980040, 040040, 040940/040 PATCH 1
2	B964P	L005968, 040040, 040940, 040040/040 PATCH 2
3	B980	L004656, 040040, 040040, 040040/040 PATCH 3
4		
5		
6		

EXHIBIT 1

OS/DOS/360 USAS COBOL CLARIFICATION

Two features of SYS/360 USAS COBOL were not documented accurately in the initial publication for these new products.

1. OS/360 USAS COBOL Compiler XREF dictionary - non-referenced items will appear in order to facilitate ease of locating item definition.

2. OS/DOS/360 USAS COBOL FILLER-condition-names (level-88 entries) may be associated with a data description entry that contains the FILLER clause so long as all others syntax rules are followed.

DOS/360 SYSIN FILES USING PUNCH FEED READ

This field contribution has not been submitted to any formal IBM test. Potential users should evaluate its usefulness in their own environment prior to implementation.

In many installations where multiprogramming is running under DOS with only one card reader, it would be nice to have a way to put a job stream out on disk for foreground batch operation. Because Job Control does not support the punch feed read for SYSIN files, we often would stop background programming to build a new SYSIN file when a job STEP was finished. This often caused a long wait when the background program was a long running program. Using S. P. I. mode of initiation of foreground programs requires too much operator console time.

Exhibit 1 shows an 80-80 list of a program which is self-relocating and reads cards on punch feed read using SYS005 and writes on DISK under file name 'IJSYSWK' using SYS004. The supervisor has to be generated with SYSFIL and BJF=YES. Then 2 label sets are cataloged in each partition label area for IJSYSWK and IJSYSIN with the same extent area and I. D. but using SYS004 and SYSIN respectively. Then multi-batching can be easily run with new job streams started or stopped as desired.

The requirement to use this program is that nothing be assigned to the punch feed read and a permanent assign of SYS004 to the disk for each partition. Then by cataloging the program with a phase card address of +0, and the name of FORINIT a foreground job stream is started by the following commands:

```
AR  BATCH F1
F1  // ASSGN SYS005, X'00D'
F1  // EXEC FORINIT
F1  // ASSGN SYSIN, X'193'
```

The program reads a job stream from the punch feed read until it reads a card which contains $$ in columns 1 and 2. Modifications that can be made for each installation could be to add permanent assign cards to the start of the job streams and some comment cards and a PAUSE card on the end to direct the operator to STOP the partition and remove it from task selection.

An alternate approach to this technique was published in INL 69-13. It used a macro instead of a program.

DOS/360 SYSIN FILES USING PUNCH FEED READ

```
// JOB COMPILE & CATALOG FORINIT IN SELF RELOCATABLE FORM
// OPTION CATAL
     PHASE FORINIT,+0
// EXEC ASSEMBLY
SYSIN    START 0
         BALR  4,0
         USING *,4
         OPENR IJSYSWK
         LA    1,RCARD
         ST    1,FIRSTA
         LA    1,PCARD
         ST    1,FIRSTA+4
         LA    1,IJSYSWK
         ST    1,FIRSTA+8
         LA    1,CARDIN
         ST    1,FIRSTA+12
         LA    3,=A(RCARD)
         MVC   0(16,3),FIRSTA
         LA    1,READCCW
         ST    1,ADDHOLD
         MVC   RCARD+9(3),ADDHOLD+1
         LA    1,CARDIN
         ST    1,ADDHOLD
         MVC   READCCW+1(3),ADDHOLD+1
         LA    1,PUNCHCCW
         ST    1,ADDHOLD
         MVC   PCARD+9(3),ADDHOLD+1
         LA    1,CARDOUT
         ST    1,ADDHOLD
         MVC   PUNCHCCW+1(3),ADDHOLD+1
READ     EXCP  RCARD
         WAIT  RCARD
         CLC   DSIGN,CARDIN
         BE    EOJ
         EXCP  PCARD
         WAIT  PCARD
         PUT   IJSYSWK,CARDIN
         B     READ
EOJ      CLOSER IJSYSWK
         EOJ
FIRSTA   DC    A(RCARD)
         DC    A(PCARD)
         DC    A(IJSYSWK)
         DC    A(CARDIN)
ADDHOLD  DS    F
RCARD    CCB   SYS005,READCCW
READCCW  CCW   X'C2',CARDIN,X'00',80
CARDIN   DS    CL80
DSIGN    DC    C'$$'
PCARD    CCB   SYS005,PUNCHCCW
PUNCHCCW CCW   X'09',CARDOUT,X'00',80
CARDOUT  DC    CL80' '
IJSYSWK  DTFSD BLKSIZE=80,DEVADDR=SYS004,IOAREA1=DISKOUT,IOAREA2=DISK2,X
               RECFORM=FIXUNB,TYPEFLE=OUTPUT,WORKA=YES
DISKOUT  DS    CL88
DISK2    DS    CL88
         END   SYSIN
/*
// EXEC LNKEDT
/&
```

EXHIBIT 1

DOS/360 GUIDELINES FOR WRITING $$B TRANSIENTS

This article has not been submitted to any formal IBM test. Potential users should evaluate its usefulness in their own environment prior to implementation.

B-transient programs are infrequently-used routines which are not resident in main storage and are, therefore, fetched or loaded from the core image library when needed. The B-transients occupy an area of 1200 bytes, referred to as the Logical Transient Area (LTA).

An SVC 2 instruction loads and executes a B-transient phase. A prefix of $$B to the name of a phase identifies it as a B-transient. The normal return to supervisor nucleus control is an SVC 11, but a transient program may exit by fetching another B-transient with an SVC 2. In the latter case, the calling B-transient is overlaid by the transient being fetched.

Register 1 must be loaded with the storage address of an 8 byte area, which contains the transient name, before the SVC 2 is issued. The Supervisor FETCH routine, then, has access to the name for searching the directories for the desired transient.

It should be noted that the Transient Directory is limited in size, and an excessive number of entries will reduce the overall speed of DOS. The Supervisor will be forced to search the Core Image Directory for any transients which cannot be included in the Transient Directory. When items are cataloged to the Core Image Library, the entry is added at the end of the CI Directory. When a /& card is read the Transient Directory is re-written by searching the Core Image Directory for any $$A or $$B entries. When the Transient Directory is full, the search is stopped. This problem is especially critical on a 2311 system which has room for only 144 $$A and $$B Transients.

Programs that are time dependent (TP) may be hampered by the excessive use of Transients. This should be considered before implementing any User written Transients.

I. Supervisor Call Interrupt (SVC)

An SVC is detected by microprogramming, which loads the SVC NEW PSW. The SVC interrupt processor analyzes the SVC code placed in the SVC OLD PSW by the calling program. Control is transferred to the appropriate processing routine.

The following SVC codes are used to FETCH Transients and RETURN to the USER program.

SVC 2: Fetches a B-transient. Loads a B-transient program (phase name prefix equals $$B) from the Core Image Library to the B-transient area and enters the B-transient at its load address plus 8 bytes. The storage address of the B-transient phase name must be supplied in general register 1. If register 1 does not point to a $$B transient name, an illegal SVC 2 results.

An address in general register 0 is ignored. The B-transient is loaded at the beginning address of the B-transient area. General register 15 is loaded with this address by the Supervisor and may be used by B-transients as a base register. Return may be either to the interrupted program or the highest priority program ready to run.

Only one program can use the B-transient area at a time. If the B-transient program is SVC 7 bound, another program is selected. This program becomes SVC 2 bound (waiting for the B-transient area) if it issues an SVC 2. Another program is then selected.

Note: The Supervisor may branch directly to the SVC 2 routine when fetching a B-transient. If the transient is not in the library when requested by the supervisor, the system enters the wait state.

SVC 8: Supplies the supervisory support to temporarily return from a B-transient program to the problem program. The B-transient area is not released. The task selection exit loads the problem program registers. An SVC 9 is used to return to the B-transient program.

SVC 9: Supplies the supervisory support for returning the B-transient after an SVC 8 is issued. The task selection exit loads the B-transient registers.

SVC 11: Returns from a B-transient releasing the B-transient area. SVC 11 is invalid if issued by other than a B-transient. The logical transient area is released for use by other programs or tasks. Return is to the highest priority program ready to run.

II. General Register Usage Conventions

SVC 2: Register 0 is the only register used as a parameter-passing register. Register 1 is loaded with the address of an 8 byte "DC" which contains the transient name to be FETCHed. Register 15 is initialized by the Supervisor as the transient base register.

Example:

```
•
•
user code
•
•
LA 1,=c'$$BRELID'
SR 0,0
SVC 2
     •
     •
     •
```

All register contents are saved prior to the FETCH of the transient in question.

If one transient issues an SVC 2 to FETCH a second transient, the first transient is overlaid and all registers remain intact.

SVC 8: Register 14 must be loaded with the User Routine entry address.

SVC 9: Register 0 is used to contain a LIOCS indicator if the LBRET macro is used after processing user labels.

In general, while executing in a logical transient ($$B), full use of all general registers is available. However, register 13 may contain unpredictable data if there was a need for ERP.

When the problem program issues an SVC 2, the Supervisor saves the problem program registers. These are restored when the LTA is released by an SVC 11 issued from the transient. The only exception occurs if an SVC 8 is issued before an SVC 11 releases the LTA. In this case, the original problem program registers are re-loaded at SVC 8 time for use during the execution of the user's routine. Then the problem program registers are re-saved when the SVC 9 returns to the transient. Therefore, when the SVC 11 is issued to release the LTA, the problem program registers restored at that time will contain the values present at the last problem program exit, i.e., at the time of the SVC 9.

III. General Information

1. $$ transients must be self-relocating (see program example).

2. $$B transients must have the CCB located above location 300 dec. This is necessary if a cancel condition occurs and you require the CCB contents. ($$BE0J3 overlays 300 bytes)

3. $$B transients must not exceed the 1200 byte limit of the LTA.

4. $$B transients should contain the transient name as the first 8 bytes of the phase to allow identification in a core dump. The base register will be initialized to location 0, but the entry point is at byte 8 of the LTA.

5. The upper end of the LTA may be used to pass data between multi-phase transients, as the LTA is not cleared by FETCH.

IV. $$B Transient Example

The following transient was written as an example to show one "self-relocation" method. Note that item III-2 was not adhered to. If a cancel condition occurs, with the phase, the messages 0P08 through 0P39 will contain invalid data.

This transient will print 3 lines of data on SYSLOG, following the MSG IPL COMPLETE.

```
01201 DOS IPL COMPLETE
VOLUME ID 123456
DATE 08/06/69
RELEASE NO. 021
```

Items 1 and 2 are retrieved from the system while item 3 is in the Transient phase. (The release number must be patched with each new DOS release.)

Program Example:

To "patch" this phase into your DOS Release 21 System, the following must be done.

1. RSERV $IPLRT2 from DOS Release 21
2. Patch $IPLRT2 as follows:

a. REP 002E04 with 47F0, E674
b. REP 004240 with 4560, F676, 4110, E684, 1B00, 0A02, 47F0, F23C
c. REP 004250 with 5B5B, C2D9, C5D3, C9C4

Change ESD card, col. 30-32 to 12-0-1-8-9, 11-6-9, 12-11-8

3. Catalog $IPLRT2 to DOS Release 21

```
// JOB CATAL
// EXEC MAINT
   CATALR IJBIPL
      .
      (DECK)
      .
      .
/*
// OPTION CATAL
   INCLUDE IJBIPL
// EXEC LNKEDT
/&
```

4. Catalog $$BRELID to DOS Release 21

```
// JOB CATAL
// OPTION CATAL
   INCLUDE
      .
      (DECK)
      .
      .
/*
// EXEC LNKEDT
/&
```

(IV. $$B Transient Example)

```
 2          PUNCH @    PHASE $$BRELID,+0@
 3 $$BRELID START 0
 4 BEGIN    EQU   *
 5          USING *,15                    BASE REG LOADED BY SUPVR
 6          USING COMREG,2                DSECT BASE REG
 7          DC    C@$$BRELID@             XIENT NAME
 8          B     *+6
 9          DC    X@0000@                 XIENT LEVEL %OPTIONAL¤
10 *        INITIALIZE ADCONS, CCW S, CCB S
11 *   INITIALIZE THE ADDRESSES OF THE MESSAGES TO BE PRINTED ON SYSLOG
12          L     5,MSG1AA                LOAD THE ADCON FOR MSG1 INTO
13          ALR   5,15                    REG 5. ADD THE BASE REG TO IT,
14          ST    5,MSG1AA                AND STORE BACK IN PROGRAM.
15          L     5,MSG2AA                SAME AS MSG1
16          ALR   5,15
17          ST    5,MSG2AA
18          L     5,MSG3AA                SAME AS MSG1
19          ALR   5,15
20          ST    5,MSG3AA
21 *   INITIALIZE THE ADDRESS OF THE CCB TO READ SYSRES VOL LABEL
22          L     5,VOLRDA
23          ALR   5,15
24          ST    5,VOLRDA
25 *   INITIALIZE THE ADDRESS OF THE CCB FOR SYSLOG
26          L     5,LOGWTA
27          ALR   5,15
28          ST    5,LOGWTA
29 *   INITIALIZE THE CCW ADDRESS WITHIN THE CCB FOR SYSRES
30          MVC   SAVE,VOLRD+8            THE SAVE AREA IS REQUIRED
31          L     12,SAVE                 BECAUSE OF BOUNDRY ALIGNMENT
32          ALR   12,15
33          ST    12,SAVE
34          MVC   VOLRD+8%4¤,SAVE
35 *   INITIALIZE THE CCW ADDRESS WITHIN THE CCB FOR SYSLOG
36          MVC   SAVE,LOGWT+8
37          L     12,SAVE
38          ALR   12,15
39          ST    12,SAVE
40          MVC   LOGWT+8%4¤,SAVE
41 *   INITIALIZE THE CCW STRING TO READ THE VOL ID ON SYSRES
42          LA    8,4                     SET CCW COUNT TO 4
43          SR    10,10                   ZERO OUT INDEX REG
44 RELOC    L     12,VOLCCW%10¤           CMD CODE AND DATA ADDRESS
45          ALR   12,15                   PLUS BASE REG
46          ST    12,VOLCCW%10¤           RETURN TO CCW
47          LA    10,8%10¤                INCREMENT TO NEXT CCW
48          BCT   8,RELOC                 DO 4 TIMES
49 *        MAINLINE
50 *   EXCP   VOLRD                       MACRO NOT EXPANDED INORDER TO
51          L     1,VOLRDA                RESOLVE ADDRESSES
52          SVC   0
```

(IV. $$BTransient Example) -Continued

```
 53 *    WAIT   VOLRD
 54             L      1,VOLRDA
 55             TM     2%1¤,X@80@
 56             BO     *+6
 57             SVC    7
 58             MVC    MSG1A,DASDIO+4          VOL ID TO MSG1
 59             MVC    LOGCNT,MSG1AL           MSG LENGTH TO CCW
 60             MVC    LOGADR,MSG1AA+1         MSG ADDRESS TO CCW
 61             BAL    6,WTCONSL               GO WRITE MSG
 62             COMRG                          GET COMREG ADDRESS FOR DATE
 63+* 360N-CL-453 COMRG     CHANGE LEVEL 3-0
 64+            L      1,20
 65+            SVC    33
 66             LR     2,1                     SET COMREG BASE REG
 67             MVC    MSG2A,DATE              MOVE DATE TO MSG2
 68             MVC    LOGCNT,MSG2AL           MSG2 LENGTH
 69             MVC    LOGADR,MSG2AA+1         MSG2 ADDRESS
 70             BAL    6,WTCONSL               GO WRITE MSG2
 71             MVC    LOGCNT,MSG3AL           MSG3 LENGTH
 72             MVC    LOGADR,MSG3AA+1         MSG3 ADDRESS
 73             BAL    6,WTCONSL               GO WRITE MSG3
 74             SVC    11                      EXIT TO JOB CONTROL
 75 *    EXCP   LOGWT
 76 WTCONSL     L      1,LOGWTA
 77             SVC    0
 78 *    WAIT   LOGWT
 79             L      1,LOGWTA
 80             TM     2%1¤,X@80@
 81             BO     *+6
 82             SVC    7
 83             BR     6
 84 MSG1        DC     C@VOLUME ID @
 85 MSG1A       DC     C@       @
 86 MSG2        DC     C@DATE @
 87 MSG2A       DC     C@XX/XX/XX@
 88 MSG3        DC     C@RELEASE NO. 021@      PATCH REL NO BEFORE CATAL
 89 MSG1AL      DC     H@16@
 90 MSG2AL      DC     H@13@
 91 MSG3AL      DC     H@15@
 92 VOLRD       CCB    SYSRES,VOLCCW,X@8000@
 93+* 360N-CL-453 CCB       CHANGE LEVEL 3-0
 94+VOLRD       DC     XL2@0@ RESIDUAL COUNT
 95+            DC     XL2@8000@ COMMUNICATIONS BYTES
 96+            DC     XL2@0@ CSW STATUS BYTES
 97+            DC     AL1%0¤ LOGICAL UNIT CLASS
 98+            DC     AL1%6¤ LOGICAL UNIT
 99+            DC     XL1@0@
100+            DC     AL3%VOLCCW¤ CCW ADDRESS
101+            DC     B@00000000@ STATUS BYTE
102+            DC     AL3%0¤ CSW CCW ADDRESS
```

(IV. $$B Transient Example)-Continued

```
103 VOLCCW    CCW    7,SEEKAD,X'40',6
104           CCW    49,SEEKAD+2,X'40',5
105           CCW    8,*-8,0,6
106           CCW    6,DASDIO,0,80
107 SEEKAD    DC     X'0000000000000003'        CYL 0  HD 0  REC 3
108 DASDIO    DS     CL80
109 LOGWT     CCB    SYSLOG,WTCCW,X'8000'
110+* 360N-CL-453 CCB       CHANGE LEVEL 3-0
111                  0,POSSIBLE ERROR IN SECOND OPERAND
112+LOGWT     DC     XL2'0' RESIDUAL COUNT
113+          DC     XL2'8000' COMMUNICATIONS BYTES
114+          DC     XL2'0' CSW STATUS BYTES
115+          DC     AL1(0) LOGICAL UNIT CLASS
116+          DC     AL1(4) LOGICAL UNIT
117+          DC     XL1'0'
118+          DC     AL3(WTCCW) CCW ADDRESS
119+          DC     B'00000000' STATUS BYTE
120+          DC     AL3(0) CSW CCW ADDRESS
121           DS     0D
122 *    WTCCW  CCW
123 *    THIS CCW, DEFINED AS A DS, CAUSED THE MNOTE IN THE LOGWT CCB
124 WTCCW     DS     0CL8
125 CMD       DC     X'09'
126 LOGADR    DC     X'0000000'
127 FLAGS     DC     X'00000'
128 LOGCNT    DC     X'00000'
129 SAVE      DS     F
130 MSG1AA    DC     A(MSG1)
131 MSG2AA    DC     A(MSG2)
132 MSG3AA    DC     A(MSG3)
133 VOLRDA    DC     A(VOLRD)
134 LOGWTA    DC     A(LOGWT)
135 COMREG    DSECT                             SUPERVISOR COMM REGION
136 DATE      DS     CL8
137           END    BEGIN
```

TYPE III AND IV PROGRAMS

Type III programs are those which have been submitted by one or more IBM employees. They are programs of general interest submitted for unrestricted distribution. They have met a basic set of programming and documentation standards but are not program tested in any formal fashion by the IBM Corporation. These programs were available for delivery from the Program Information Department prior to June 23, 1969.

Type IV programs are those contributed for unrestricted distribution by one or more authors of which at least one is an employee of an IBM customer. They are made available by IBM essentially in the author's original form, but conform to published Type IV standards. IBM exercises no control over the technical content of the documentation but merely assures that the quality of reproduction is satisfactory. Type IV programs have not been tested by IBM.

IBM makes no warranty, expressed or implied, as to the documentation, function or performance of these programs, and the user is expected to make the final evaluation as to their usefulness in his own environment.

---------- TYPE III REVISION ----------

360D-06.7.011 FILING AND SOURCE DATA ENTRY TECHNIQUES FOR EASIER RETRIEVAL (FASTER-OS). Material Revised - The Documentation and Machine Readable.

BASIC PROGRAM PACKAGE
DOCUMENTATION - Write-up.
MACHINE READABLE - Appropriate material delivered.

OPTIONAL PROGRAM PACKAGE - None.

ORDERING INFORMATION: PROGRAM NUMBER 360D067011

	PROGRAM NUMBER EXTENSION	DISTRIBUTION MEDIUM TYPE	CODE	USER VOLUME REQUIREMENT
BASIC	none	DTR 9/800	28	none
		DTR 9/1600	29	none
		DTR 7DC/800	26	none
OPTIONAL	none	none		none

---------- TYPE III REVISION ----------

360D-06.3.003 BASIC AUTODIN ACCESS METHOD FOR S/360 DOS. Material Revised - The Documentation, Machine Readable and Catalog Abstract.

ORDERING INFORMATION: PROGRAM NUMBER 360D063003

	PROGRAM NUMBER EXTENSION	DISTRIBUTION MEDIUM TYPE	CODE	USER VOLUME REQUIREMENT
BASIC	none	DTR 9/800	28	none
		DTR 9/1600	29	none
		DTR 7DC/800	26	none
OPTIONAL	none	none		none

---------- TYPE III REVISION ----------

360D-06.7.012 FILING AND SOURCE DATA ENTRY TECHNIQUES FOR EASIER RETRIEVAL (FASTER-DOS). Material Revised - The Documentation and Machine Readable.

BASIC PROGRAM PACKAGE

DOCUMENTATION - Write-up.
MACHINE READABLE - Appropriate material delivered.

OPTIONAL PROGRAM PACKAGE - None.

ORDERING INFORMATION: PROGRAM NUMBER 360D067012

	PROGRAM NUMBER EXTENSION	DISTRIBUTION MEDIUM TYPE	CODE	USER VOLUME REQUIREMENT
BASIC	none	DTR 9/800	28	none
		DTR 9/1600	29	none
		DTR 7DC/800	26	none
OPTIONAL	none	none		none

---------- TYPE IV REVISION ----------

360D-16.4.010 SYNTH1 (A COMPUTER PROGRAM FOR THE DESIGN OF REACTIVE NETWORKS FROM A SET OF ARBITRARY POLES AND ZEROES). Material Revised - The Documentation.

---------- TYPE IV REVISION ----------

360D-16.2.020 COLUMN DESIGN - WORKING STRESS METHOD. Material Revised - The Documentation and Machine Readable.

ORDERING INFORMATION: PROGRAM NUMBER 360D162020

	PROGRAM NUMBER EXTENSION	DISTRIBUTION MEDIUM TYPE	CODE	USER VOLUME REQUIREMENT
BASIC	none	DTR*	00	none
OPTIONAL	none	none		none

---------- NEW TYPE IV PROGRAM ----------

GC-21

DESCRIPTION - This program allows calculation of the displacements, internal moments and reactions of a plate of any shape in bending. The plate can be supported in any possible way and loaded by any external force. The plate can be of constant or variable thickness. It can be simply supported or clamped on separate points or along segments of straightline. The value of the displacement at some points can also be imposed; this possibility is used in the analysis of the subsidence of a support. The external forces are formed by distributed vertical loads, concentrated loads, concentrated moments applied to the boundary. Several load cases can be processed in the same run. The mathematical model of Kirchhof is used; in this model, the unknown is the vertical displacement of the middle plane of the plate. The numerical calculation is then performed by the finite element method. The curved boundary is treated as polygonal lines; the plate is divided into triangular and rectangular elements; both types can be used together in a same computation.

PROGRAMMING SYSTEMS - Written in FORTRAN IV; operates under OS/360.

MINIMUM SYSTEM REQUIREMENTS - The program runs on a S/360 Model 40 with 128K minimum. Modifications to the program require a 256K machine for re-compilation.

BASIC PROGRAM PACKAGE
DOCUMENTATION - Write-up.
MACHINE READABLE - Object code, source code and sample problem.

OPTIONAL PROGRAM PACKAGE - None.

ORDERING INFORMATION: PROGRAM NUMBER 360D162022

	PROGRAM NUMBER EXTENSION	DISTRIBUTION MEDIUM TYPE	CODE	USER VOLUME REQUIREMENT
BASIC	none	DTR 9/800	28	none
		DTR 9/1600	29	none
		DTR 7DC/800	26	none
OPTIONAL	none	none		none

---------- NEW TYPE IV PROGRAM ----------

COMPUTER CALCULATION OF FREQUENCY RESPONSE FUNCTIONS FROM STEP RESPONSE DATA.

DESCRIPTION - Subroutine BODE was constructed to determine the frequency response of a system whose transient response to a step function is known. The program is based on a procedure developed by Samulon which includes phase and magnitude correction factors. Frequency response field testing requires that a frequency source be applied to the system under study and magnitude and phase shift determined for each individual frequency of interest. Through the use of this program, it is necessary to apply only a step change to the system in the field and record its transient response. The transient response amplitude at discrete points is fed into the program along with the time interval between each data point. The program supplies as its output the desired frequency response data.

PROGRAMMING SYSTEMS - Written in FORTRAN IV.

MINIMUM SYSTEM REQUIREMENTS - Those associated with the FORTRAN IV compilers.

BASIC PROGRAM PACKAGE
DOCUMENTATION - Write-up.
MACHINE READABLE - Source code and sample problem.

OPTIONAL PROGRAM PACKAGE - None.

ORDERING INFORMATION: PROGRAM NUMBER 360D168002

	PROGRAM NUMBER EXTENSION	DISTRIBUTION MEDIUM TYPE	CODE	USER VOLUME REQUIREMENT
BASIC	none	DTR*	00	none
OPTIONAL	none	none		none

---------- NEW TYPE IV PROGRAM ----------

1400 INPUT/OUTPUT STANDARD LABEL PROCESSING EXITS FOR 483 DISK & TAPE SORT

DESCRIPTION - These two exit routines are designed to process 1400 input and output standard tape labels under the DOS Tape & Disk Sort/Merge (360N-SM-483). Multifile and Multireel processing is supported, but deblocking is not supported. Mixed 360 standard labels and 1400 standard labels are supported on input. Label errors are handled under the control of computer operator via the console typewriter. The routines must be cataloged in the user's Core Image Library. They are self-relocating, therefore, able to operate in any batched partition with the SORT program. This routine should be of value to installations having large core and using 1400 emulation. It will allow them to take advantage of the increased speed and foreground capabilities of the 483 Sort when working with 1400 tapes.

PROGRAMMING SYSTEMS - The exits are written in Assembler language and were assembled under DOS Release 18. The exits were tested under DOS Release 19.

MINIMUM SYSTEM REQUIREMENTS - System/360 Model 30 and any hardware required by DOS/360.

BASIC PROGRAM PACKAGE
DOCUMENTATION - Write-up.
MACHINE READABLE - Source code.

OPTIONAL PROGRAM PACKAGE - None.

ORDERING INFORMATION: PROGRAM NUMBER 360D003028

	PROGRAM NUMBER EXTENSION	DISTRIBUTION MEDIUM TYPE	CODE	USER VOLUME REQUIREMENT
BASIC	none	DTR 9/800	28	none
		DTR 9/1600	29	none
		DTR 7DC/800	26	none
OPTIONAL	none	none		none

---------- NEW TYPE IV PROGRAM ----------

KWIC INDEX GENERATOR

DESCRIPTION - This package contains three mainlines and the associated subroutines to generate a Key-Word-In-Context index. Input format is completely general, and will probably accept cards which have already been punched for another purpose. Program operation is controlled through questions and answers on the console keyboard and typewriter.

PROGRAMMING SYSTEMS - Written in FORTRAN, and the three parts of the mainline are linked through Call Link statements.

MINIMUM SYSTEM REQUIREMENTS - Requires an 8K machine with 1132 Printer, 1442 Card Read-Punch, and one disk.

BASIC PROGRAM PACKAGE
DOCUMENTATION - Write-up.
MACHINE READABLE - Source code and sample problem.

OPTIONAL PROGRAM PACKAGE - None.

ORDERING INFORMATION: PROGRAM NUMBER 1130066017

	PROGRAM NUMBER EXTENSION	DISTRIBUTION MEDIUM TYPE	CODE	USER VOLUME REQUIREMENT
BASIC	none	CARDS	15	none
OPTIONAL	none	none		none

---------- NEW TYPE IV PROGRAM ----------

PURDUE FARM SUPPLY CENTER MANAGEMENT GAME

DESCRIPTION - The Purdue Farm Supply Center Management Game simulates the competitive market environment in which farm supply centers selling feeds, fertilizers and other agricultural inputs and purchasing grains compete. From one to eight teams can compete directly in a market (set). Each team (Center) makes 41 decisions which would normally be made on an annual basis such as price levels for products and services, quantity and quality discounts, hog and layer

contracting, personnel, investment, facility expansion and orders. The game can be used to teach business planning techniques, economic and accounting principles or nature of the firm and industry.

PROGRAMMING SYSTEMS - Written in FORTRAN IV and contains 10 subroutines. Read and write statements have been written in variable form for easy adaption to other computer equipment.

MINIMUM SYSTEM REQUIREMENTS - An 1130 8K Disk System with a 1442 Card Reader/Punch and a 1403 Printer. Unused core is 600 (HEX) words.

BASIC PROGRAM PACKAGE
DOCUMENTATION - Write-up.
MACHINE READABLE - Source code and sample problem.

OPTIONAL PROGRAM PACKAGE - None.

ORDERING INFORMATION: PROGRAM NUMBER 1130155001

	PROGRAM NUMBER EXTENSION	DISTRIBUTION MEDIUM TYPE	CODE	USER VOLUME REQUIREMENT
BASIC	none	CARDS	15	none
OPTIONAL	none	none		none

---------- NEW TYPE IV PROGRAM ----------

MOON FLIGHT

DESCRIPTION - Plan your own flight to the moon. Starting from a parking orbit, add an incremental velocity at some point around the orbit, then coast. Loops and figure-eights around the moon are possible. Velocity corrections, permitted at any time, make it possible to orbit the moon and return to earth. The flight may be traced by listing the result of every tenth iteration of the solution on the printer. An after-the-fact plot of the trajectory may be dumped on the printer. For users with on-line plotting devices, a call to a subroutine 'LTPLT' makes available the 'X-Y' coordinates (in miles) after iteration.

PROGRAMMING SYSTEMS - Written in FORTRAN; operates under 1130-OS-005 (Disk Monitor, Version 2).

MINIMUM SYSTEM REQUIREMENTS - 1130 Model 2B (I/O is user's option).

BASIC PROGRAM PACKAGE
DOCUMENTATION - Write-up.
MACHINE READABLE - Object and source code.

OPTIONAL PROGRAM PACKAGE - None.

ORDERING INFORMATION: PROGRAM NUMBER: 1130302005

	PROGRAM NUMBER EXTENSION	DISTRIBUTION MEDIUM TYPE	CODE	USER VOLUME REQUIREMENT
BASIC	none	CARDS	15	none
OPTIONAL	none	none		none

---------- NEW TYPE IV PROGRAM ----------

COMPUTER PROGRAM FOR A HOSPITAL BLOOD BANK

DESCRIPTION - Computer program for maintaining records for a hospital blood bank.

PROGRAMMING SYSTEMS - Written in 1800 FORTRAN and 1800 Assembler language and uses 1800 TSX. Requires MIOF (available from DACS Centers) and MAGOP (1800-03.4.001) Type 3 programs available from IBM, Program Information Department.

MINIMUM SYSTEM REQUIREMENTS - 16K 1801, 1-1810, 1-1442, 1-1443, and a 1-1816.

BASIC PROGRAM PACKAGE
DOCUMENTATION - Write-up.
MACHINE READABLE - Object code and sample problem.

OPTIONAL PROGRAM PACKAGE - None.

ORDERING INFORMATION: PROGRAM NUMBER 1800252001

	PROGRAM NUMBER EXTENSION	DISTRIBUTION MEDIUM TYPE	CODE	USER VOLUME REQUIREMENT
BASIC	none	MT 9/800	28	01
		MT 9/1600	29	01
		MT 7DC/800	26	01
OPTIONAL	none	none		none

ACCUMULATIVE LIST OF NEW TYPE III AND IV PROGRAMS

Following is an accumulative list of new Type III and Type IV programs which have been made available since the last edition of the Catalogs of Programs or their Supplements. Abstracts describing these programs can be found in the Installation Newsletter Issue No. shown below.

Program Number	Program Title	Newsletter Issue No.
360D-00.0.017	SPOOL, A PACKAGE TO CONVERT CURRENT OR FUTURE PRINTED REPORTS TO MAGNETIC TAPE FOR FOREGROUND OR BACKGROUND PRINTING LATER	69-09
360D-00.0.020	2495 UTILITY PROGRAM FOR OS/360	69-11
360D-00.0.022	PUFER - PUNCH FEED READ UTILITY CARD TO TAPE	69-15
360D-00.1.021	S/360 GENERALIZED EXTRACTION AND MATCH PROGRAM	69-07
360D-00.1.022	NYU PRINT-UTILITY PROGRAM	69-07
360D-00.2.006	OS SYSTEM/360 FLOWCHART (OS FLOW): DOS FLOWCHART UNDER OS/360	69-03
360D-00.3.025	S/360 DOS TAPE TO PRINTER/PUNCH SPOOLING SYSTEM	69-08
360D-00.3.026	FORTRAN TAPE DISK READ WRITE SUBROUTINES	69-16
360D-00.3.027	SEVEN TRACK OCTAL TAPE DUMP	69-11
360D-00.3.028	1400 INPUT/OUTPUT STANDARD LABEL PROCESSING EXITS FOR 483 DISK & TAPE SORT	69-19
360D-00.4.015	DAP DIRECT ACCESS PATCH PROGRAM	69-09
360D-00.4.018	IBM SYSTEM/360 INDEXED SEQUENTIAL DASD TO PRINT UTILITY	69-17
360D-00.5.005	FORMATTED LISTING OF VTOC FOR OS/360	69-08
360D-00.5.006	SHAREDISK	69-11
360D-00.6.011	A HYPERTEXT EDITING SYSTEM FOR THE S/360 USING THE 2250 DISPLAY	69-18
360D-01.0.009	OS/360 DATE AND TIME SUBROUTINE (DT01) WITH CONVERSION FROM JULIAN DATE TO GREGORIAN	69-09
360D-01.1.006	COMPILE AND GO FOR THE OS 360, PL/I (F) COMPILER	69-06

360D-01.3.002	1287 CHECK DIGIT AND NUMERIC RECONSTRUCTION MACROS	69-11
360D-01.4.010	SYSGEN AIDE: DIRECTORY DUMP	69-16
360D-03.0.007	S/360 MOD 44 MULTIPROGRAMMING SYSTEM	69-08
360D-03.0.008	DOS IOCS LOGIC MODULE SOURCE STATEMENT GENERATOR	69-08
360D-03.0.010	STENO TO ENGLISH TRANSLATION	69-17
360D-03.0.012	PARS CONTROL (ACP)	69-13
360D-03.1.014	FAST ASSEMBLER - INTERPRETER FOR Sytem/360 ASSEMBLER LANGUAGE PROGRAMS (VERSION 1)	69-14
360D-03.2.011	OS/360 PL360 COMPILER	69-06
360D-03.2.013	BASIC COMPILER UNDER TSS/360	69-13
360D-03.2.014	THE SIMSCRIPT II PROGRAMMING LANGUAGE	69-16
360D-03.2.015	THE XPL COMPILER GENERATOR SYSTEM	69-17
360D-03.3.008	COURSE III/OS A REAL-TIME COMPUTER ASSISTED INSTRUCTIONAL SYSTEM	69-01
360D-03.3.009	1404 IOCS MACROS FOR DOS/TOS	69-09
360D-03.3.010	SNAP PROCESS OR (PROTOTYPE)	69-10
360D-03.3.011	COMIT/360	69-15
360D-03.4.026	BSAM SUBROUTINE PACKAGE FOR OS 360, VERSION I	69-06
360D-03.4.027	FORTRAN RANDOM I/O SUBROUTINE	69-03
360D-03.4.028	2260 LOCAL TERMINAL USERS SYSTEM (LOTUS)	69-06
360D-03.4.031	KNUTH PROPOSAL INPUT/OUTPUT PACKAGE (KIOP)	69-16
360D-03.6.009	OS/360 COBOL CROSS-REFERENCE AND RESEQUENCING PROGRAM	69-05
360D-03.6.010	JOB STREAM MANAGER OS/MVT	69-03
360D-03.6.012	PROGRAM FOR THE CREATION OF CROSS REFERENCE TABLES, FROM S/360 RPG SOURCE PROGRAMS, FOR INDICATORS AND FIELD NAMES	69-10
360D-03.6.013	SYSTEM MANAGEMENT ROUTINES OS/MVT	69-10
360D-03.6.017	PL/60	69-15
360D-03.6.018	NEATER: A PL/I SOURCE STATEMENT REFORMATTER	69-17
360D-03.7.025	CARRIER RETURN/LINE FEED ON THE 1052/53, MISLCRLF	69-03
360D-03.7.026	DCLC A DOS/TOS/OS ASSEMBLER MACRO TO DEFINE A LOWER-CASE CHARACTER CONSTANT - DEFINE-CONSTANT-LOWER-CASE	69-09
360D-03.7.027	EDIT MACRO FOR DOS, TOS, OR OS ASSEMBLY LANGUAGE, S/360	69-07
360D-03.7.028	DOS MACRO FOR TYPEWRITER INPUT/OUTPUT	69-10
360D-03.7.031	DOS MACROS TO DUMP ISAM FILES FROM 2311 OR 2314 TO TAPE AND RESTORE FROM TAPE TO 2311 OR 2314	69-15
360D-03.8.013	PL/I STRING FUNCTIONS	69-06
360D-03.8.015	A PLAN FOR A LINKED-DIRECT FILE, WITH OPERATING ROUTINES, IN COBOL	69-13
360D-04.0.008	DOS COBOL CROSS REFERENCER	69-04
360D-04.0.009	PCMSG - PROGRAM CHECK MEyage	69-09
360D-04.1.010	DEBUGGING AID FOR DOS PROGRAM TESTING	69-04
360D-04.1.011	S/360 POST MORTEM DUMP (PCP/MFT)	69-09
360D-04.2.007	PROGRAM TO AID IN OPTIMIZING COMPUTE BOUND JOBS BY INDICATING THE MAJOR PROCESSING LOOPS	69-04
360D-04.6.001	A PROGRAM FOR GRAPHIC ON-LINE DEBUGGING ON AN IBM 2250 DISPLAY	69-08
360D-05.1.016	CONVERSATIONAL REMOTE BATCH ENTRY (CRBE)	69-13
360D-05.2.010	INTERACTIVE APPLICATIONS SUPERVISOR (IAS/360)	69-06
360D-05.2.011	DOS/360 TIME ACCOUNTING SYSTEMS	69-10
360D-06.1.005	MARC II SORT PROGRAM	69-16
360D-06.3.006	GENERALIZED TELEPROCESSING PROGRAM FOR SYSTEM/360	69-07
360D-06.3.007	DOS BTAM FOR LEASED LINE KSR 35 TELETYPES	69-06
360D-06.3.008	AUTODIN ACCESS METHOD FOR OS/360	69-10
360D-06.5.006	UNIVAC-1108 TO IBM-360 FLOATING POINT INTERNAL CONVERTER ('CVFL08')	69-06
360D-06.5.007	7090 TO 360 FLOATING POINT INTERNAL CONVERTER ('CVFL09')	69-06
360D-06.6.003	FORTRAN CHARACTER STRING PACKAGE	69-03
360D-06.6.004	CHARACTER FILTER PL1	69-03

360D-06.7.018	BSEARCH - A RANDOM ACCESS BINARY-SEARCH TECHNIQUE FOR SEQUENTIAL FILES ON DISK OR DRUM	69-03
360D-06.7.019	KWADE-KEYWORD AS A DICTIONARY ENTRY	69-03
360D-06.7.021	AUTORAF - A SYSTEM FOR BUILDING AND USING RECORD ADDRESS FILES FOR DIRECT PROCESSING OF FILES ON SYSTEM/360	69-10
360D-06.7.024	GENERATION DATA GROUP (GDG) MODIFICATION	69-10
360D-07.6.001	2741 READ AND WRITE ROUTINE	69-06
360D-08.0.003	WRIMAT MATRIX WRITER	69-05
360D-10.0.001	PROGRAM TO CALCULATE THE AMOUNT OF CPU TIME AVAILABLE IN THE BACKGROUND WHEN RUNNING DOS MULTIPROGRAMMING	69-03
360D-10.0.002	FORTRAN ANALYSIS PROGRAM	69-04
360D-10.2.004	HIGH WATER MARK (HWM)	69-09
360D-10.2.005	CORE AND/OR RESOURCE CONFLICT (CRC) NOTIFICATIONS	69-10
360D-10.2.006	NUCLEUS INITIALIZATION PROGRAM (NIP) MODIFICATIONS	69-11
360D-11.1.023	DISK RESIDENCE FOR 1410 COMPATIBILITY MODE PROGRAMS ON THE IBM S/360 MODEL 40	69-10
360D-12.1.020	PL/I BINARY TO EBCDIC CODE KEEPING FIVE BINARY CODED COLUMNS PRINTING RECODED DATA	69-11
360D-12.2.008	CDC - 3800/3300 COBOL TO IBM - S/360 COBOL TRANSLATOR	69-09
360D-12.2.009	NCR 315 TRANSLATOR FROM NEAT TO 360 ALP	69-09
360D-13.0.004	GRAPHICS STEPWISE REGRESSION ROUTINE (BMDGR1)	69-15
360D-13.0.005	NON-LINEAR LEAST SQUARES ON GRAPHICS (BMDGR2)	69-16
360D-13.0.006	TIME SERIES SPECTRUM ESTIMATION (BMDGT1)	69-15
360D-13.0.009	HISTOGRAM SUBROUTINE	69-16
360D-13.1.005	GENERAL TAXONOMY PROGRAM	69-08
360D-13.2.003	LEAST-SQUARES ESTIMATION OF NON-LINEAR PARAMETERS - NLIN	69-09
360D-13.4.007	RESCALED RANGE TRANSFORM	69-06
360D-13.4.008	COLLEY-TUKEY FAST FOURIER TRANSFORM -- FOUR 2 IN ASSEMBLER LANGUAGE	69-17
360D-13.7.001	DIALL - GENERAL LEAST SQUARES DIALLEL ANALYSIS OF VARIANCE	69-15
360D-15.4.006	ICES/360 SOURCE PROJECT-I	69-05
360D-16.0.001	UCARDS: UNION CARBIDE AUTOMATIC ROUTINE AND DESIGN SYSTEM FOR PRINTED CIRCUIT BOARDS	69-15
360D-16.2.018	BULKHEAD SHEET PILING ANALYSIS	69-06
360D-16.2.019	ICES/360 STRUDL-II	69-09
360D-16.2.020	COLUMN DESIGN - WORKING STRESS METHOD	69-09
360D-16.2.021	ICES/360 SOURCE TRANSET I	69-14
360D-16.2.022	GC-21	69-19
360D-16.2.023	ICES/360 COGO-I SOURCE	69-16
360D-16.2.024	ICES/360 ROADS*I*SOURCE	69-16
360D-16.2.025	ICES/360 TABLE-II, OBJECT AND SOURCE	69-16
360D-16.3.001	ECONOMIC EVALUATION AND SCREENING PROGRAM FOR RESEARCH PROJECTS	69-05
360D-16.4.009	LISA 360 A PROGRAM FOR LINEAR SYSTEMS ANALYSIS	69-09
360D-16.4.010	SYNTH1 (A COMPUTER PROGRAM FOR THE DESIGN OF REACTIVE NETWORKS FROM A SET OF ARBITRARY POLES AND ZEROES)	69-13
360D-16.8.002	COMPUTER CALCULATION OF FREQUENCY RESPONSE FUNCTIONS FROM STEP RESPONSE DATA	69-19
360D-17.7.001	SUBROUTINE UCONIC	69-07
360D-19.5.008	MVT ACCOUNTING ROUTINE (MVTACTRT)	69-12
360D-19.7.010	PROOF OF DEPOSIT ENTRY RUN WITH TRANSIT DISTRIBUTION	69-08
360D-19.7.011	PERSONAL TRUST ACCOUNTING PACKAGE TO ASSIST USERS IN INSTALLING OFF-LINE PROCESSING	69-07
360D-23.1.003	TWO-STAGE, TWO-DIMENSIONAL TRIM PROGRAM II	69-10
360D-25.0.004	EQUIPMENT ORDER, SHIPMENT, INSTALLATION, AND DISCONTINUANCE REPORT FOR LARGE ACCOUNTS	69-08

360D-25.2.004	UNIT CONTROL SYSTEM FOR RETAILERS	69-08
360D-29.3.002	VSP SCHEDULE PRODUCTION FOR O/S 360	69-16
360D-30.1.002	360/2260 FUNCTIONAL DEMONSTRATION IN DOS F1 PARTITION - BTAM LOCAL	69-09
360D-30.2.002	SHARED HOSPITAL ACCOUNTING SYSTEM (SHAS) DEMONSTRATION PACKAGE	69-09
360D-40.4.003	MULTIPLE - PRECISION FLOATING-POINT ARITHMETIC PACKAGE	69-11
360D-42.2.001	EXPERIMENTAL PROGRAM FOR DETERMINING POLYNOMIAL ZEROS	69-06
360D-45.1.001	INVRSOL	69-03
360D-45.2.001	EIGENRS	69-03
360D-99.0.004	/360 DOS INDIVIDUALIZED HOMEWORK ASSIGNMENT AND QUIZ GENERATOR (QUIZ AND EXERCISE DEMONSTRATOR: QED)	69-05
360E-00.0.008	SIMPLIFIED USER ORIENTED REPORT - WRITING PROGRAM FOR 8K MODEL 20	69-05
360E-01.0.001	INVALID CCCP/360-20 INVALID CODE CARD CORRECTION PROGRAM	69-07
360E-01.0.002	FIELD SELECTION AND LIST	69-08
360E-01.5.001	MODEL 20 STAND-ALONE DISK COPY UTILITY	69-07
360E-03.4.002	SHORT INPUT OUTPUT SYSTEM	69-06
360E-06.5.002	JULIAN DATE PROCESSING SUBROUTINES	69-08
360E-06.6.003	CONVERTING A NUMBER TO IT'S VERBAL DESCRIPTION (CKPR)	69-16
360E-13.0.001	HOSPITAL PBAR PROGRAMS FOR A MFCM MODEL 20	69-07
360E-13.0.002	STATISTICAL COMPARISONS PROGRAM	69-07
360E-13.1.006	SINGLE DIGIT COUNT AND FREQUENCY DISTRIBUTIONS	69-12
360E-13.6.001	LINEAR REGRESSION IN TWO VARIABLES (LEAST SQUARE FIT) FOR THE MOD 20	69-06
360E-16.4.004	MINIMIZATION OF BOOLEAN EXPRESSIONS	69-16
360E-16.5.001	CALCULATE MASTER CAMS	69-11
360E-19.7.020	SAVINGS ACCOUNTING FOR THE 360 MOD 20 DISK	69-04
360E-23.1.003	HEURISTIC PAPER MACHINE TRIM PROGRAM FOR THE IBM 360/20	69-10
360E-25.2.006	DOPTIM - DISK ORDER POINT TECHNIQUE OF INVENTORY MANAGEMENT	69-18
360E-27.3.001	AMERICAN FLYERS AIRLINE AVAILABILITY REPORT	69-13
360E-28.3.013	360/20 CARD DESCRIPTIVE BILLING	69-09
360E-28.3.014	WHOLESALE AND/OR RETAIL, ACCOUNTS RECEIVABLE, AGED TRIAL BALANCE	69-06
1130-00.0.009	SFREF - FORTRAN AND ASSEMBLY LANGUAGE SUBPROGRAM, LINK, AND FILE READ/WRITE CROSS-REFERENCE TABLE BUILDING PROGRAM	69-11
1130-00.3.010	1130 PRNZ MODIFIED FOR ADDITIONAL CARRIAGE CONTROL	69-07
1130-00.4.007	1130 DISK-PACK SECURITY PROGRAM	69-16
1130-01.6.003	ISFMS - 1130 INDEXED SEQUENTIAL FILE MANAGEMENT SYSTEM ON THE 1130	69-05
1130-03.4.011	REREAD FOR FREE FORMAT CARD READING IN FORTRAN	69-10
1130-05.5.001	SECAL - A PROGRAM FOR BUILDING MULTIPLE LOCAL SUBROUTINE MODULES FOR THE IBM 1130	69-10
1130-06.3.001	IBM 1130 REMOTE JOB ENTRY SYSTEM	69-06
1130-06.3.002	WILLIAM AND MARY REMOTE JOB ENTRY PROGRAM - 8K DISK SYSTEM VERSION (WMRJ8)	69-11
1130-06.3.003	WILLIAM AND MARY REMOTE JOB ENTRY PROGRAM - 4K DISK AND CARD SYSTEMS VERSION	69-11
1130-06.6.014	THE 1130/1800 KEYWORD INDEX GENERATOR	69-04
1130-06.6.015	IBM 1130 FORTRAN REREAD DEVICE AND FREE FORMAT KEYWORD INPUT ROUTINES	69-06
1130-06.6.016	ALPHAMERIC DATA STATEMENT PRODUCER FOR FORTRAN	69-09
1130-06.6.017	KWIC INDEX GENERATOR	69-19
1130-06.7.003	IBM 1130 INDEX FOR DM2 SYSTEMS	69-11
1130-07.0.002	RASEC/RDSEC	69-11

1130-08.0.002	HIGH SPEED AND OVERLAP 1132 PRINTER ROUTINES FOR 1130 FORTRAN AND BAL USERS	69-16
1130-12.1.005	1130 FILE CONVERSION PROGRAM, DISK MONITOR VERSION I TO DISK MONITOR VERSION 2 (FCOPY)	69-10
1130-13.7.001	MULTIPLE RANGE TEST FOR CORRELATED AND HETEROSCEDASTIC MEANS	69-08
1130-15.5.001	PURDUE FARM SUPPLY CENTER MANAGEMENT GAME	69-19
1130-15.2.005	1130 BLENDING OF INITIAL FURNACE CHARGES	69-10
1130-15.4.007	W.E.L. CHART SCHEDULING PROGRAM	69-16
1130-17.3.002	GENERAL PURPOSE CHROMATOGRAPH PEAK INTEGRATION PROGRAM	69-07
1130-17.6.002	1130 RAW SCORE TEST ANALYSIS PROGRAM	69-15
1130-17.6.003	1130 TEST ITEM ANALYSIS PROGRAM	69-16
1130-17.6.004	APGRD A PROGRAM TO GRADE AND ANALYZE MULTIPLE CHOICE QUIZZES	69-18
1130-21.1.001	IBM 1130 HOSPITAL PAYROLL	69-09
1130-23.3.002	COMPUTER APPROACH TO INVENTORY REQUIREMENTS ON THE 1130 FOR SMALL ASSEMBLY ORIENTED MANUFACTURERS	69-09
1130-28.2.002	1130 PATIENT BILLING	69-08
1130-28.3.001	1130 HOSPITAL ACCOUNTS RECEIVABLE	69-08
1130-30.1.004	INDIVIDUALIZED HOMEWORK ASSIGNMENT AND QUIZ GENERATOR (QUIZ AND EXERCISE DEMONSTRATOR: Q.E.D.)	69-04
1130-30.2.003	BATTLE OF THE NUMBERS	69-12
1130-30.2.004	1130 PERPETUAL CALENDAR DEMONSTRATION PROGRAM	69-13
1130-30.2.005	MOON FLIGHT	69-19
1800-00.0.002	OBJECT TIME I/O LOGICAL NUMBER GENERATOR	69-04
1800-00.0.003	CARD UTILITY PROGRAM USING 1800-OS-010, MPX	69-10
1800-01.0.002	BOM CORE & DISK DUMP UTILITIES MODIFIED TO RUN IN A CORE LOAD AREA OF V-CORE FOR USE WITH 1132 OR 1443 LINE PRINTERS	69-11
1800-17.4.001	REDUCTION OF OCEANOGRAPHIC STATION DATA	69-09
1800-23.5.006	IBM ANALOG DATA ACQUISITION PROGRAM	69-04
1800-23.5.008	PROCESS DATA PREPARATION PROGRAM	69-10
1800-25.2.001	COMPUTER PROGRAM FOR A HOSPITAL BLOOD BANK	69-19
1401-10.3.071	1401 UNIVERSITY ADMISSIONS INFORMATION SYSTEM	69-14
1401-10.3.072	FINANCIAL AID INFORMATION SYSTEM (LOAN SUBSYSTEM)	69-14
1401-10.3.073	FINANCIAL AID INFORMATION SYSTEM (STUDENT AID SUBSYSTEM)	69-14
1620-01.6.160	LABELR - A REPORT GENERATOR	69-15
1620-02.0.066	A MODIFIED PDQ FORTRAN SYSTEM	69-16
7040-H2 3552	MULTIDIMENSIONAL OPTIMIZATION USING PATTERN SEARCH	69-11
7094-F2 3554	EIGENVALUE - EIGENVECTOR ROUTINE FOR REAL COMPLEX MATRIX	69-15

IBM Installation Newsletter

October 10, 1969 Issue No. 69-20

CONTENTS

(G means Green Section)

Distribution: Branch Office-DP Management, Salesmen, Systems Engineers, FE Managers. Regions, Districts, Education Centers, Field Systems Centers, Federal Systems Centers, FE Area Offices, DPD HQ, FED HQ, WTC.

*Requires Immediate Attention

OS/360 RELEASE 18 MVT DEDICATED WORKFILES IMPROVE PERFORMANCE

All OS/360 MVT installations should consider using the Release 18 Dedicated Workfiles facility. The results of measurements indicate an improvement in throughput of a jobstream using dedicated workfiles over a jobstream using non-dedicated workfiles. The facility is most helpful when the jobstream has many steps that require temporary direct access space because neither allocation nor unallocation of the temporary space is done at step time.

Runs were made on an OS/360 Model 50I using two 2311's for system residence and two 2311's for scratch space. For this configuration, the use of the Dedicated facility gave between a 1.1 and a 1.4 second improvement per dedicated data set used when compared with non-dedicated data sets. Measurements were made on a pre-Release 18 system containing some Release 17 modules.

Since a typical compile-link edit-go (CLG) job requires temporary data sets, there can be a savings of several seconds per job.

An analysis of the runs yields the following table:

Improvement in Scheduler Time - CLG

	COBOL	FORTRAN	PL/I	Utility
Reduction in Scheduler Time	24%	12%	25%	55%
Improvement (sec.) per dedicated data set used	1.2	1.2	1.3	1.4

If three minutes were spent in the execution (CLG) of each job, then the projected savings in minutes per hour for each initiator would be: COBOL-2.0, FORTRAN-0.9, PL/I-1.9, and Utility-4.7.

The Release 18 Dedicated Workfile facility was written for MVT only, since a user of PCP has the capability to pre-allocate data sets. When an initiator with the Dedicated facility is started, specific data sets are allocated for all the jobs running under it and are unallocated when the initiator is stopped. Those jobs that desire a dedicated data set obtain it by using the DSNAME key word recognized by the initiator. There is no contention between initiators for a dedicated data set because each initiator allocates different space for its own dedicated data sets. Therefore, jobs running under different initiators may use the same DSNAMEs without any confusion.

OS/360 PERFORMANCE OF MULTIPLE INITIATORS

How many initiators should be started with MVT? This question has troubled several accounts. Normally, there are limits to the number of initiators caused either by the amount of main storage available or by the number of I/O devices available. But large systems such as the models 85, 91, and the multiprocessing M65 may not be so limited by these considerations. A more fundamental point relates to the design of the system itself. Are there any limitations in the system that would reduce the effectiveness of a large number of initiators? These notes will discuss such a limitation in the system and the kinds of things a user might choose to do about it.

Upon first glance there does not seem to be any limitation on the number of initiators other than the fifteen protect keys. Since the reader and writers operate in key zero they are outside this limitation. However, there is a function in MVT that can hamper the effectiveness of a large number of initiators. This function is the allocation/termination routine.

Allocation/termination code is serial in nature. It uses the ENQ macro to guarantee that only one initiator at a time can have access to the UCB's. Therefore, with any jobstream there is a limit on the speed that the jobstream will run regardless of an increased number of initiators. This limit is the time necessary to run allocation/termination twice for each jobstep in the stream.

Let us look briefly at some of the things that happen in allocation. In the first place the queue records for the job step must be read from and written into SYS1.SYSJOBQE. Device allocations are made. For direct access volumes (and tapes as well if AVR is included) mount messages are issued and the mounting of the correct volume is checked. For new direct access data sets the actual space that the data set will occupy is assigned at this time. Finally, messages are written into the SYS1.SYSJOBQE data set describing the allocations that have taken place.

It is possible to measure the amount of time that is spent waiting for allocation as well as the time in allocation. This can be done by modifying the ENQ routines to check for the name of the allocation resources (SYSIEFSD Q4 and Q5). These measurements were done on a Model 91 with several jobstreams, a commercial stream and a scientific stream. The ground rules for the measurement were that no writers were active and no mounts were necessary because PRESRES was used to insure that all needed volumes were pre-mounted.

Chart 1 gives the percentage of time that an average task spent waiting for allocation or termination. Chart 2 gives the time for the jobstream with various numbers of initiators as well as the time that Q4 and Q5 were in use. Notice that in one case increasing the number of initiators caused the time for the entire jobstream to increase. It is also worth noticing that the case when SYS1.SYSJOBQE was located on a 2311 shows that an average task spent from 30-60% of its time just waiting for allocation or termination. These measurements do not include a 2314 SYS1.SYSJOBQE but we can speculate that the 2314 would be somewhere between the 2301 and the 2311 on Chart 1. It would probably be closer to the 2311 than to the 2301.

Clearly, the case of eight initiators shown in Chart 2 demonstrates that the allocation routines are in use 100% of the time for the 2311. Even on the 2301 the usage is over 75% of the total time.

It is true that the jobstreams used were fairly high intensity i.e., many short job steps. Remember, though, that no mounts took place during the measurement nor was there a writer active. These functions would contribute to Q5 usage. However, there are going to be customers who wish to run high-intensity job streams on our large systems. The next section will explore what the user might do in case the allocation/termination routines seem to be a bottleneck in his applications.

Recommendations

1. Large systems should consider placing SYS1.SYSJOBQE on the 2301. This action seems reasonable when more than four or five initiators will be used to schedule jobs. Notice that initiators that are tied up for long periods with a single job step such as a TP job can be ignored in this connection.

2. Since space assignement for new data sets on direct access takes place during allocation, the number of new data sets should be held to a minimum. Creation of a new data set can take several seconds and hence lengthen allocation time considerably. A Release 18 improvement will help substationally in this respect. This incremental will allow an initiator to have Dedicated Work Files that could be used by the job steps that are selected by that initiator. In the absence of this feature, it may be possible to use class scheduling and dedicated work files. However, this would normally mean that only one job of a given type could be running at a time.

3. Reducing the number of jobsteps in a job will obviously reduce the number of times that the allocation/termination routines must execute. For this reason the Loader would seem to offer much help to certain environments. One step monitors that use the ATTACH capability might also be investigated for some applications. For small volume sorts the PL/I SORT feature and the COBOL SORT VERB would reduce scheduler usage.

4. Careful planning needs to be done to reduce the impact of mount time. The time that is spent by an operator searching for a volume involves a complete lockup in allocation. When this lockup occurs all the writers will pause when they finish processing a data set. The PUBLIC/PRIVATE facilities of the system should be used where possible to eliminate unnecessary mounting and dismounting of volumes. For example, if a PRIVATE volume must be used in more than one step of a job the "RETAIN" parameter should be used to prevent an unnecessary mount/dismount. Class scheduling should be used to balance the necessary setup so that the operator can keep up and delays are avoided.

In addition, there was a Scheduler Change distributed by Poughkeepsie HQ FSC to all SD&I Centers on July 25, 1969. This change applies to Release 17. It will allow the scheduler to leave allocation without waiting for a mount of an "old" data set. This change should reduce the time spent in allocation. This local modification has been approved for distribution by SD&I Centers as an exception to a general policy.

5. The use of Automatic Volume Recognition (AVR) must be studied to determine whether or not it is justified. The reason why AVR is a consideration relates to its effect on the processing of tape volumes. Without AVR the mount of a tape is not required during allocation; instead the mount can be deferred until OPEN time. When AVR is included in the system, tape volumes must be mounted before the initiator leaves allocation. If the operators are able to stay ahead by premounting tape volumes then there will be no delay. On the other hand, if the operators are not able to keep ahead, it might be better to remove AVR from the system.

6. There is not much the user can do to reduce the dependence of the writer on Q5. Hopefully, some measures suggested will reduce the contention for allocation and thus reduce the impact on the writer. There are no plans to solve this currently.

7. A final possibility that might be looked at in MVT is track stacking (INITQBUF). In some cases track stacking may reduce the time spent in allocation because when the stack is written back to SYS1.SYSJOBQE the initiator can leave allocation. Without track stacking individual 176 byte records would be written instead. However, overall activity into SYS1.SYSJOBQE may increase because instead of a single record being read or written an entire stack must always be read. Selecting an optimum value for track stacking may prove difficult.

Conclusion

The potential effects of allocation/termination on multiple initiator performance have been discussed in this report. Several methods have been explored to reduce this impact but any complete solution of the problem must await changes in the logic of the operating system.

Chart 1

Percentage of time that an average task spent waiting for Q4 or Q5, Model 91.

Number of Initiators	JOBQE on 2311	JOBQE on 2301
4	32%	4%
5	44%	7%
6	52%	8%
7	58%	12%
8	64%	16%

Chart 2

Number of Initiators	JOBQE on 2311		JOBQE on 2301	
	Total Time Sec.	Q5 Time Sec.	Total Time Sec.	Q5 Time Sec.
5	820	810	850	520
6	820	810	880	600
7	800	770	825	580
8	840	840	800	660

Note: In the system with JOBQE on the 2311 LINKLIB and FORTLIB were on 2301.

OS/360 CHECKPOINT TIMINGS REPORT

Introduction

The HQ FSC has recently completed a series of measurements of the OS/360 Checkpoint facility under MVT. The purpose of these measurements was not to study all possibilities in the use of Checkpoint. Instead, it is intended to provide some planning information for those accounts that are going to install Checkpoint under MVT. Those who want the essential results of the study should turn to the section of this report entitled "Conclusions".

Background

The Checkpoint/Restart feature of OS/360 is new in Release 17. We ran a series of measurements on a model 50 with MVT to determine what the performance of an individual checkpoint is and what the impact of checkpoint on a production job might be. We do not claim that our measurements are universally applicable nor did we do an exhaustive set of measurements.

The method used to derive the timings for an individual checkpoint was to precede and follow the execution of the checkpoint SVC with a request for the time. In the absence of any other active tasks in the system this measure of elapsed time is used to determine performance.

Elapsed time is also used as the measurement of sort performance. Sort was chosen as the vehicle to measure the effect of Checkpoint on a production type run. We felt that OS Sort was the component of the operating system where most accounts would be likely to implement checkpoint first.

Care was taken during the measurements that we made to printout LOGREC and observe any errors that might have occurred. In this way we were able to guarantee that I/O errors did not invalidate our measurements.

As a final point it is worth noting that the Checkpoint data set was never OPEN when we issued the CHECKPOINT macro. Timings would be improved if this data set were OPENed by the problem program.

Conclusions

1. Elapsed time for an individual checkpoint has been established as from 4-15 seconds in the cases studied. Table 1 gives details on these measurements.

2. By observation most of the checkpoint time seems to be I/O time. The measurements are probably roughly applicable to either a Model 40 or a Model 65. However, our systems residence was a 2314. We did not measure what a change here would do to timing.

3. Users will want to increase the blocking factor on the checkpoint data set up to the maximum. When the checkpoint data set is on a 2314 the maximum blocksize produces a checkpoint time three times as fast as the minimum blocksize. However, when the checkpoint data set is a tape, users will want to consider the time involved for error recovery if extremely large blocksizes are used.

4. When the checkpoint data set is on the 2314 the use of chained scheduling in our runs did not produce significant improvement over full track blocking.

5. The times for these checkpoints could be reduced further if the checkpoint data set were OPEN before the checkpoint macro was issued.

6. Table 2 shows the impact of checkpoint on a five minute 2314 sort when the Balanced merge technique was used. Notice that the user should specify DISP=MOD if his checkpoint data set is on tape. This is contrary to the example on page 54 of the Sort SRL (C28-6543-5). The increase in Sort time shown in Table 2 could be reduced by specifying a larger blocksize on the checkpoint data set.

7. Table 2 also shows that suppressing the checkpoint through the JCL is as effective so far as timing is concerned as removing the checkpoint macro from the problem program.

8. Table 3 shows an overhead for checkpoint of about 25 seconds in a total sort time of 530 seconds without checkpoint. This case also shows that the effect of increasing the blocksize on the checkpoint data set was a saving of nearly a minute in a ten minute sort.

9. By comparing Table 2 and Table 3 we can compare the checkpoint protection of the oscillating tape sort and the balanced (2314) sort for this application. As far as number of checkpoints taken the two sort techniques are nearly equivalent, five for balanced and six for oscillating. The frequency of checkpoints differs. With the 2314 sort a checkpoint was taken on an average of once a minute while, with the tape sort, a checkpoint was taken on an average of once every two minutes. A faster tape drive would presumably change this ratio. Howver, users will have to evaluate this consideration in their own environment.

HARDWARE CONFIGURATION

All measurements were made on a 512K model 50. Tapes are on Channel 2. All tapes were 9 track 90KB. System residence was on a 2314 on Channel 2 with a two channel switch so that it could be accessed through Channel 3 as well.

OPERATING SYSTEM

The operating system used in the measurements was Release 17 MVT. The RSVC, BLDL and RAM module listings are given in Table 4, 5 and 6 respectively. Pertinent options from the Stage I Sysgen are given below.

```
CTRL PROG TYPE=MVT, MAXIO=50,
FETCH=PCI, OPTIONS=ROLLOUT, QSPACE=10
OVERLAY=ASYCHRON
SCHEDULR TYPE=MVT, ACCTRTN =
SUPPLIED, OPTIONS = (BYLABEL, RJE),
VLMOUNT=AVR, JOBQFMT = 15, JOBQLMT =
150, JOBQTMT = 60

SUPRVSOR RESIDENT = (ATTACH,
EXTRACT, IDENTIFY, SPIE, BLDLTAB,
RENTCODE, TRSVC), OPTIONS =
(IDENTIFY, TRSVCTBL, PROTECT,
COMM, ONLNTEST, RER), TIMER =
JOBSTEP, TRACE=5, SER=SER 1
```

CHECKPOINT TIMING

MODEL 50
SYSRES-2314

Location of Checkpoint Data Set	70K	Region 100K	Size 200K
90KB Tape BLKSIZE = 600	6.05 sec.		12.53 sec.
90KB Tape unspecified BLKSIZE	4.20 sec.		7.25 sec.
2314 BLKSIZE = 600	7.17 sec.		15.27 sec.
2314 FT Blocking	4.03 sec.		5.13 sec.
2314 FT Blocking-(OPCD=C, NCP-2)	4.27 sec.		
2314 FT Blocking		4.27 sec.	
2314 FT Blocking-(OPCD=C, NCP=2)		4.20 sec.	
2314 FT Blocking-(OPCD=C, NCP=3)		4.25 sec.	
2314 FT Blocking-(OPTCD=C, NCP=10)		4.20 sec.	
2314 BLKSIZE = 600		8.62 sec.	
2314 BLKSIZE = 600 (OPTCD=C, NCP=5)		6.67 sec.	

NOTE: 1. Checkpoint data set not open when CHKPT Macro issued.

2. In all cases CHKPT Macro issued at EOV exit from DCB exit list.

TABLE 1

IMPACT OF CHECKPOINT ON SORT

MODEL 50 - REGION SIZE 200K
SYSRES - 2314

50,000 records	LRECL = 100
Sortin, Sortout on 90KB Tape	
BLKSIZE = 5,000	3 work areas on 2314
Checkpoint Data Set on 90KB Tape	
	Sort Time
Checkpoint Data Set DISP=KEEP, BLKSIZE = 600	6 min. 55 sec.
Checkpoint Data Set DISP=MOD, BLKSIZE = 600	5 min. 42 sec.
RD=RNC (No checkpoints taken)	4 min. 46 sec.
RD=RNC & CHKPT parameter removed from Sort Card	4 min. 46 sec.

Note: 5 Checkpoints taken during sort.

TABLE 2

IMPACT OF CHECKPOINT ON SORT

MODEL 50 - REGION SIZE 200K
SYSRES - 2314

50,000 records	LRECL = 100
Sortin, Sortout on 2314	BLKSIZE = 7,000
4 90KB 2400 work tapes	
Checkpoint Data Set on 2314	
	Sort Time
Checkpoint Data Set BLKSIZE = 600	10 min. 6 sec.
Checkpoint Data Set No blocksize specified	9 min. 15 sec.
RD=RNC (Suppress checkpoint by JCL)	8 min. 50 sec.

Note: 6 Checkpoints taken during sort.

TABLE 3

BLDL MODULE LIST - TABLE 4

IEEVATT1	IEFSD080	IEFW21SD
IEEVWAIT	IEFSD085	IEFW41SD
IEEVWTR1	IEFSD086	IEFW42SD
IEFIRC	IEFSD087	IEFXJ000
IEFQMRAW	IEFSD094	IEF085SD
IEFQMSSS	IEFSD104	IEF086SD
IEFSD060	IEFVHA	IEWL
IEFSD061	IEFVHN	IEWLF128
IEFSD062	IEFVH1	IEWLF440
IEFSD065	IEFVM1	IEWLF880
IEFSD071	IEFWC000	IEWSZOVR
IEFSD079	IEFWD000	LINKEDIT

RAM MODULE LIST - TABLE 5

IGG019CK	IGG019CA	IGG019BC
IGG019AV	IGG019CB	IGG019AL
IGG019AM	IGG019AD	IGG019BD
IGG019AH	IGG019AN	IGG019BE
IGG019AG	IGG019AC	

RSVC MODULE LIST - TABLE 6

IGC00021	IGG0550I	IGG0201A
IGC0003B	IGG0550L	IGC0002%
IGC0003E	IGG0550M	IGG0191A
IGC0005E	IGG0550N	IGG0191B
IGC0002D	IGG0550Z	IGG0191D
IGC0002A	IGG0551A	IGG0191I
IGC0002H	IGG0325B	IGG0191J
IGC0001I	IGG0325D	IGG01910
IGC0003D	IGG0325E	IGG01911
IGC0007B	IGG0325G	IGG0190I
IGC0107B	IGG0325H	IGG0190L
IGG0CLF2	IGG0290A	IGG0190M
IGG0553A	IGG0290B	IGG0190S
IGG0553B	IGG0290C	IGG0190Y
IGG0553C	IGG0290D	IGG0190Z
IGG0553D	IGG0200A	IGG0191N
IGG0553E	IGG0200F	IGG0200Y
IGG0550K	IGG0200G	IGG0290E

OS/360 PUBLICATIONS GO MICROFICHE

DP is conducting a pilot Microfilm Program on the IBM System/360 Operating System (OS) Publications, excluding the Program Logic Manuals (PLM).

Presently, fifty-seven (57) of the SRL Publications are available on Microfiche from the IBM Distribution Center.

Each 3-1/4 x 7-3/8" (Tab Card size) Microfiche card contains up to ninety (90) 8-1/2 x11 text pages, reduced twenty-four times (24:1).

Associated TNLs have been incorporated into their respective base manuals resulting in an up-to-date Microfiche card.

The Microfiche version will carry a form number similar to the hard copy version except the second character in the prefix will now be an alphabetic character; whereas the suffix will now become zero (0) on the initial Microfiche card version regardless of the hard copy form number suffix. Example:

Hard copy version form number and title

C28-6631-6 IBM System/360 OS Messages and Codes

Microfiche version form number and title

CB8-6631-0, IBM System/360 OS Messages and Codes

In the above example, CB8-6631-0 is equivalent to C28-6631-6 and includes a subsequent up-date, TNL N28-2373.

Both of the above form numbers will appear in the various catalogs and indexes available.

Use key "G" applies to both versions.

The Microfiche card is in addition to, and not intended to replace the hard copy version of the SRL publications.

S/360 Model 25 PRINT SPEED COMPARISONS USING MCS ON NATIVELY ATTACHED 1403 CLARIFICATION

An article on this subject appeared in INL 69-18. Reader response indicates the subject may still need further clarification. The article has been modified and is again printed below.

When configurating or planning for the use of the Multiple Character Set feature on a 1403 natively-attached to a 2025, particular attention must be paid to possible degradation in expected print speeds. Perferred character arrangements will run at the slowest of the speeds specified for that arrangement in the type catalog. To avoid severe print speed degradation, it is desirable to choose an alternate comparable "straight" arrangement over a preferred arrangement, whenever possible. A "straight" arrangement is one in which each graphic appears the same number of times on the chain/train.

Following is a chart of preferred arrangements with comparable straight arrangements and speed comparisons for a 1403-N1.

NOTE: There is no lozenge on the PN arrangement. If a lozenge is required by an RN user, it may be obtained by substituting a lozenge in each array for any character not needed by the customer.

For further information consult the Type Catalog of the Sales Manual.

Arrangement	Speed
PCS-AN, PCS-HN	Numeric, Alpha, or Special-560 LPM
Alternate AN, HN	All Graphics-1100 LPM
QN, QNC	Alpha, Numeric, or Specials-310 LPM
Alternate PN	All Graphics-955 LPM
RN	Alpha, Numeric, or Specials-310 LPM
Alternate PN (Note)	All Graphics-955 LPM
SN	Alpha, Numeric, or Specials-310 LPM
Alternate TN	All Graphics-560 LPM
YN	Alpha, Numeric, or Specials-560 LPM
Alternate AN or HN	All Graphics-1100 LPM

S/360 Model 20 BSCA IOCS Modification

Problem: Unit record error on 360/M20 causing the communications line to drop.

Problem Description: When operating with leased line logic on a switched network, a unit record error on the S/360 Model 20 will cause the line to drop. (Ex. Reader Stop, Forms Check, etc.) Interrupt Handler Routine gains control when the error occurs and passes control to the BSCA IOCS ABORT routine which will issue a 'CIO DISABLE BSCA' command. The resultant effect is that the data set will go 'ON HOOK,' terminating the telephone connection before the operator can clear the unit record error.

Circumvention: To avoid the disconnecting of the line:

1. Locate the S/360 Model 20 BSCA IOCS ABORT Module and verify the statement after label IDK9 which reads:

```
9B5300C8  C10 200, X '53' DISABLE BSCA
replace with:
470000C8                    NOP
```

2. Continue issuing Read or Write Initials until the operator has cleared the unit record error and continued.

S/360 Transmitting to Model 20:

With this correction applied the ABORT routine will be entered as before, but only an EOT will be sent to S/360. Consequently, when S/360 is the transmitting station the user can cycle on a write TI after encountering an EOT response. DOS BTAM will post user with X'41' and bit 1 of BYTE 24 (DECB FLAG BYTE) will be on if it is an EOT.

The Model 20 BSCA IOCS Routine will eventually return control to the user. BYTE 2, Bit 0 of the TECB will indicate whether there was an ABORT or not. If so, the user can cycle on a READ dtfname, TR. When the I/O error is corrected, S/360 and the model 20 can now get back into SYNCH and resume processing.

S/360 Receiving From Model 20:

The above solution applies with the only difference being that S/360 will cycle on a READ TI and the Model 20 will issue a write TW.

2711 LINE ADAPTER UNIT SALES MANUAL PAGE CORRECTION

Item four under Specify states Use of Cable Order Form 120-1080 is not applicable to the 2711.

Use Communications Cable Order Form 120-1507 when ordering cables. The Sales Manual page will be corrected. In the meantime, correct the existing page.

INFORMATION ABOUT THE NEWSLETTER

EVERY SALESMAN AND SYSTEMS ENGINEER SHOULD RECEIVE ONE COPY OF THE NEWSLETTER. Each FE Branch Manager should receive five copies for distribution to Customer Engineers.

The IBM Installation Newsletter is distributed only to IBM locations and is not available to customers. The Newsletter is separated into two parts. The first part contains information which is intended only for IBM personnel and the pages are labeled For IBM Internal Use Only. THESE PAGES, OR REPRODUCTIONS OF THEM, ARE NOT TO BE GIVEN TO CUSTOMERS. However, using your own good judgement, you may discuss pertinent information from these pages with a customer. The second part contains information which may be reproduced by the Branch Office at their discretion and given to customers. These pages are not labeled with any restrictive classification. Note that when any abstracts for Type III and Type IV programs are given to the customer, he must be informed that the disclaimers for Type III and Type IV programs as contained in the Newsletter, applies. YOUR COOPERATION IN PROPERLY HANDLING THE CONTENT OF THE IBM INSTALLATION NEWSLETTER IS NECESSARY FOR ITS CONTINUED PUBLICATION.

Changes to the quantity of the Newsletter shipped are to be made by the Administration Manager. He should update and send the DP Administration Managers' Document Control card (Electro IBM N61508) to DP Document Control, DPD HQ.

Direct inquiries concerning the contents of this Newsletter to:

W. F. Malmborg, Editor
IBM Installation Newsletter
Dept. 812, DPD HQ
112 East Post Road
White Plains, New York 10601
Telephone 914 949-1900 Extension 6158
Tie Line 374-6158

NEW PUBLICATIONS ANNOUNCED BY PRLs

Information is extracted and condensed from the weekly PRLs (Publications Release Letters) to insure that all Salesmen and Systems Engineers are aware of new or revised Reference Sources, Marketing Publications, and Field Engineering Publications. Normally, each issue of the Newsletter will contain information extracted and condensed from two PRLs, one following the other. The information will be placed in the Newsletter in its original sequence with no rearrangement of form numbers or titles. It is not intended to replace existing information and distribution sources. You should be certain that you are aware of these sources.

Do not use the condensed Newsletter information to order publications. Most but not all (for example, some films and slide sets) of the items are available from Mechanicsburg. To receive the items you need without delay, prior to ordering check the REQ (Requisition Form) code in the "Accumulative Supplement to IBM Publications Current Price List (Z20-0100)" which is attached to the weekly Publications Release Letter (PRL) and distributed to your Administration Manager. For an explanation of the REQ codes and other publications information see the first few pages of the "IBM Publications Current Price List Z20-0100".

The IBM publications requisition (MO2-0618) or ITPS should be used for ordering items from the IBM Distribution Center, Mechanicsburg, Pa.

(IBM World Trade Corporation has its own distribution center and procedures.)

PRL #38 September 19, 1969

MARKETING PUBLICATIONS

FORM NO.	TITLE	DISPOSITION
221-0384-2	IBM 2803/2804 Tape Control Model 1 Proposal Insert	NEW
320-1029-0	Purdue Agricultural Accounting Program	NEW
320-1030-0	SHAS - Terminal Operators Guide	NEW
320-1906-8	August 1969 "In Brief"	Scrap 320-1806-7
520-2196-0	IBM System/3 for Banks	NEW
520-2206-1	System/360 Model 195	NEW
548-0012-0	CE Career Guide	NEW
548-0013-0	CE Manager Career Guide	NEW
549-0717-0	MT/SC Third Tape Station Specification Folder	NEW
A27-3013-0	IBM 2770 Data Communication System System Components	NEW
A33-4500-1	IBM S/360 Component Description - 2826 Paper Tape Control, 1017 Paper Tape Reader, 1018 Paper Tape Punch	Scrap A33-4500-0
C20-1646-5	A Programmer's Introduction to IBM S/360 Assembler Language Student Text	Scrap C20-1646-4
C30-2020-0	IBM S/360 OS Planning for Telecommunications Access Method	NEW
E20-0333-0	Library Automation - Computer Produced Book Catalog	NEW
K20-0322-0	Bankmark Charge Card System - Optical Character Reading and Descriptive Billing Using the IBM S/360 and 1287 Optical Reader	NEW
K20-0330-0	Consolidated Functions Processing IBM S/360 Model 20 with Direct Access Storage at United Home Life Insurance Company Application Brief	NEW
N23-0684-0	TNL to Introduction to S/360 Advisor Guide Re: R29-0244	NEW
N27-1316-0	TNL to S/360 OS and 1130 Disk Monitor System - S/360 - 1130 Data Transmission for FORTRAN Re: C27-6937-1	NEW
N33-9060-0	TNL to S/360 Model 20 DPS Systems Generation and Maintenance - C33-6006-2	NEW
R20-1037-0	S/360 Disk And Tape OS Coding (DOS/TOS) Education Guide	NEW
R20-1039-2	S/360 Disk and Tape Operating System Coding (DOS/TOS) Student Materials	NEW
R20-1078-2	S/360 Operators Reference Guide Handbook	UES R20-1078-1
R20-6003-0	System/3 Application Design Education Guide	NEW
R20-8083-0	S/360 Model 20 Installation Programming - Disk/Tape Learner - Paced Format Study Guide	NEW
R20-8084-0	S/360 Model 20 Installation Programming - Disk/Tape Learner - Paced Format Student Materials	NEW
R20-8085-0	S/360 Model 20 Installation Programming - Disk/Tape Learner - Paced Format Advisor Guidelines	NEW
R20-8086-0	IBM 1800 Data Acquisition and Control System MPX Conversion Education Guide	NEW
R20-9001-2	IBM 1401 Data Processing System Accelerated Basic Programming Course Description	Scrap R20-9001-1
R20-9171-1	S/360 Assembler Language Coding Workshop Course Description	NEW
R20-9293-0	S/360 Model 20 Installation Programming - Disk/Tape Learner-Paced Format Course Description	NEW
R20-9306-0	System/3 Application Design Course Description	NEW
R20-9309-0	System/3 Executive Introduction Course Description	NEW
R29-0211-4	System/360 COBOL Writing Programs in Cobol Reference Handbook	UES R29-0211-3
R29-0253-1	System/360 Model Assembler Language Coding Sample Problems	UES R29-0253-0
X20-1786-0	Numerical Control Coding Form/AUTOSPOT Tool Information Form	NEW
Y20-0044-2	Index Tabs for DP Sales Manual	Scrap Y20-0044-1

PRL #38 (Continued)

Y20-0431-0	Information Management System/360 for the IBM System/360 System Manual Volume 1 - Program Logic	NEW
Y20-0432-0	Information Management System/360 for the IBM S/360 System Manual, Volume 2 - Flowcharts	NEW
Y20-0458-0	TNL to Service for Consultants Re: Y20-0161	NEW
Y33-6003-1	PL/I Language Specifications	Scrap Y33-6003-0
Y33-8019-0	TNL to S/360 Model 20 Remote Job Entry Work Station Program Logic-Y33-8006	NEW
Y33-9044-0	TNL to S/360 Model 20 DPS Control and Service Program Logic Manual Re: Y33-9008-2	NEW
YBO-0433-0	Information Management S/360 for the S/360 System Manual, Volume 3 - Assembly Listings	NEW
Z77-9192-0	Software Measurement of the Performance of a Tele-communications Based System	NEW

FIELD ENGINEERING PUBLICATIONS

123-1028-1	2265 Display Station Parts Catalog	Scrap 123-1028-0
225-6369-6	557 Alphabetic Interpreter FEMM	UES 225-6369-5
229-0033-0	FES to IR, MINI-IR, PAIR Preparation Instructions Booklet Re: 229-0021-0	NEW
R23-3104-1	2701 Data Adapter Unit Type1 Student Self Study	Scrap R23-3104-0
R23-3128-2	2313 Disk Storage Drive Study Course	Scrap R23-3128-1
R23-3157-1	2841 Storage Control Student Self-Study Course	Scrap R23-3157-0
R27-9506-2	S/360 Channel Adapter Student Self Study	UES R27-9506-1
R25-5424-0	Selectric I/O Mechanical Keyboard	NEW
R25-5425-0	2025 ICA Start/Stop Adapter Student Guide	NEW
R31-0281-1	1907 Batch Recording Model 5	Scrap R31-0281-0
R31-0455-0	1907 Model 6 Rots Mark Scanner Student Guide	NEW
S29-2545-0	Supplement to 2948 FE Education Student Guide Re: R23-3127-0	NEW
S29-3557-0	Supplement to 514 519 Student Guide Re: R31-0313-0	NEW
Y22-6608-0	2365 Processor Storage Models 1, 2, 3 and 13 FETOM	NEW
Y22-6753-0	208S Processing Unit:E-Unit Operations FETOM	NEW
Y22-6755-0	S/360 Model 85 Channel Adapter FETMM	NEW
Y22-6762-0	208S Base Machine FEMDM	NEW
Y22-6763-0	208S Features FEMDM	NEW
Y25-2768-1	2841 Storage Control Student Quiz #1	Scrap Y25-2768-0
Y25-2771-1	2841 Storage Control Student Quiz #4	Scrap Y25-2771-0
Y25-2772-1	2841 Storage Control Prerequisite Quiz	Scrap Y25-2772-0
Y25-3059-2	1130 Computing System Student Quiz	Scrap Y25-3059-1
Y25-3783-1	84 Sorter Prerequisite Quiz	Scrap Y25-3783-0
Y27-2138-2	2848 Display Control FETOM	Scrap Y27-2138-1
Y33-3003-0	1018 Paper Tape Punch FEMM	NEW
Z31-0220-0	5496 Data Recorder Preliminary FETOM	NEW

PRL #39 September 26, 1969

REFERENCE SOURCES

FORM NO.	TITLE	
N20-0360-61	System/360 SRL Newsletter Re: A22-6822-13	Scrap N20-0360-60
N20-1130-38	1130 SRL Newsletter Re: A26-5916-4	Scrap N20-1130-37
N20-1869-02	TNL to BAP Insurance Bibliography Re: H20-0536-0	Scrap N20-1869-1

MARKETING PUBLICATIONS

221-0369-3	IBM 2250 Proposal Insert	Scrap 221-0369-2
221-0401-3	IBM 2203 Printer, Models A1 and A2 Proposal Insert	Scrap 221-0401-2
221-0405-3	IBM 2560 Proposal Insert	Scrap 221-0405-2
504-1012-0	The Programmers - Recruiting Brochure	NEW
504-4116-13	Employment Fact Sheet	Scrap 504-4116-12
520-1081-4	System/360 Model 20 Facts Folder	UES 520-1081-3
520-2142-1	IBM 2314 Direct Access Storage Facility	Scrap 520-2142-0
570-0427-0	IBM System/3 - 96 Column Punched Card Sample	NEW
CC0-2003-0	IBM S/360 OS QTAM Message Processing Programming Services - C30-2003-3	NEW
CC0-2005-0	IBM S/360 OS QTAM Message Control Programming - C30-2005-2	NEW
CC0-2006-0	IBM S/360 OS RJE - C30-2006-2	NEW
CC0-2008-0	IBM S/360 OS Planning for RJE MFT Version II - C30-2008-0	NEW
CC0-2010-0	IBM S/360 OS Planning for Conversational RJE - C30-2010-0	NEW
F20-8172-6	Bibliography of Data Processing Techniques	UES F20-8172-5
H20-0340-3	Structural Engineering System Solver for the IBM 1130 Version User's Manual	UES H20-0340-2
H20-0532-1	System/360 Power System Planning for Electric Utility Industry Application Description Manual	Scrap H20-0352-0
H20-0683-0	APL/360 User's Manual Program No. 5734-SM1 and 5736-SM1	NEW
H20-0684-0	APL/360 OS Operations and Installation Manual	NEW
H20-0685-0	APL-360 DOS Operations Manual	NEW
H20-0686-0	APL/360 DOS System Generation Manual	NEW
H20-0689-0	APL/360 Primer Program Number 5734-XM1 and 5736-XM1	NEW
K20-0329-0	Online Data Entry, Authorization, Inquiry and Updating Using IBM 2260	NEW

PRL #39 (Continued)

N20-2050-0	TNL to Coursewriter III for S/360 Version 2 Author's Guide Re: H20-0609-0	NEW
N23-0687-0	TNL to 1401 DPS Basic Programming SPS Re: R29-0044-2	NEW
N24-0437-0	TNL to S/360 Model 30 1401/1440/1460 Compatibility Feature Re: A24-3255-7	NEW
R20-9165-1	IBM S/360 Assembler Language Coding - PI Course Description	Scrap R20-9165-0
R20-9224-1	IBM 1130 Computing System Continuous System Modeling Program Course	Scrap R20-9224-0
R20-9307-0	IBM S/3 Installation Control for Management Course Description	NEW
R20-9308-0	IBM System/3 Introduction to Computing Systems Course Description	NEW
R29-0143-2	IBM Punched Card Data Processing Principles 83 Sorter Operation Examination	UES R29-0143-1
R29-0265-1	IBM Basic PL/I Coding Additional Coding Techniques Text and Examination	UES R29-0265-0
V20-4619-0	Laboratory Automation Based Systems	NEW
V20-4626-0	Improved Data Entry with the IBM 50 Magnetic Data Inscriber and 2495 Magnetic Tape Cartridge Reader	NEW
V20-4628-0	The Number Game	NEW

		DISPOSITION
V20-4630-0	It Works and That's Key	NEW
V20-4631-0	Transition to Tomorrow	NEW
V20-4634-0	Operating System/360...The Most for Your Money	NEW
V20-4635-0	Paper 'Making It' Today	NEW
V20-4636-0	Engineering Data Processing...More Power to Engineers	NEW
V20-4637-0	IBM System/3	NEW
V20-4638-0	Oil-Challenge and Change	NEW
V20-4639-0	The Advanced Administrative System	NEW
V20-4640-0	IBM System/3 Concept and Design	NEW
V20-4641-0	The Achievers	NEW
V20-4642-0	Breakthrough	NEW
Y19-0002-0	Requirements Planning and Inventory Control System Manual	NEW
Y20-0104-1	1130 Work Measurement Aids, Version 2 System Manual, Volume 1	Scrap Y20-0104-0
Y20-0169-0	The World of OCR	NEW
Y20-0450-0	TSS/360 Time Sharing Compendium	NEW
Y20-0463-0	Production Information and Control System Executive Presentation - Foil/ Script	NEW
YBO-0470-0	APL/360 OS Assembly Listings: System Manual Program No. 5734-XM1	NEW
YBO-0471-0	APL/360 DOS Assembly Listings: System Manual Program No. 5736-XM1	NEW

FIELD ENGINEERING PUBLICATIONS

229-6002-0	IR/PAIR Code Guide - IBM 2420 Tape Drives	NEW
BOF-5319-0	2701 System Data Adapter Unit 1 - A22-6864, R25-5431, Y27-2062 & Y27-2064	NEW
R27-9605-0	2020 P U Submodel 2 & 4 BSC Adapter Student Guide	NEW
R31-0205-8	CDP Planning Chart	Scrap R31-0204-7
R31-0282-1	1907/1935 Model 5 Batch Record System Course Material	Scrap R31-0282-0
R31-0359-2	Bill of Forms Processing Requisite	UES R31-0359-1
R31-0456-0	1907/1784 Batch System Supplementary Course Material	NEW
R31-0461-0	5496 Data Recorder Student Guide	NEW
S25-0505-0	FES to S/360 Model 25 External Field Definitions Handbook Re: 229-2176	NEW
S29-2544-0	FES to Microfiche Cards V31-0060-1	NEW
V31-0118-1	S/360 Model 25 I/O Microfiche Card	Scrap V31-0118-0
Y25-2226-0	2025 ICA Start/Stop Adapter Student Quiz	NEW
Y25-2785-1	2701 Data Adapter Unit Type I Student Quiz	Scrap Y25-2785-0
Y25-3144-0	2020 PU Submodel 2 and 4 BSC Adapter Student Quiz	NEW
Y25-3145-0	2020 PU Submodel 2 and 4	NEW
Y25-3156-0	1403 Printer Mechanics Student Quiz 1	NEW
Y25-3703-3	84 Sorter Student Quiz	Scrap Y25-3703-2
Y25-3848-3	Basic Non-Technical Orientation Student Guide	Scrap Y25-3848-2
Y25-3897-0	General Systems Technical Orientation Student Quiz	NEW
Y25-3899-0	2540 Student Quiz 2	NEW
Y25-3900-0	2540 Reader Student Quiz #3	NEW
Y25-3901-0	2540 Punch Student Quiz #2	NEW
Y25-3903-0	1403 Printer Student Quiz #2	NEW
Y25-3904-0	1403 Printer Student Quiz #3	NEW
Y25-3909-0	1403 Printer Student Quiz #1	NEW
Y27-0014-3	2740 Model 1, 2740 Model 2 and 2741 Communication Terminal FEMDM	Scrap Y27-0014-2 & Y27-1037

October 10, 1969

IBM INSTALLATION NEWSLETTER

Issue No. 69-20

Table of Contents

DOS/360 COMPILERS, ASSEMBLER AND LINK EDITOR IN FOREGROUND

This field contribution has not been submitted to any formal IBM test. Potential users should evaluate its usefulness in their own environment prior to implementation.

An alternate approach for Assembler and Link Editor was described in INL 69-10.

Job Control, Assembler F, Linkage Editor, and maintenance functions (except condense of core image libraries) can run in Foreground Two or Foreground One as well as background with a few simple changes.

These changes were made to DOS Release 20 and operate on a 128K Model 40 with a 16K Supervisor. Excessive turnaround time experienced by the programmers prompted these changes.

Changes only involved six modules: IJBJC3, IJBJC2, IJBJC1, IJBLE1, IJSCBL01, and IJR000.

First, we consider Job Control. We have three objectives to accomplish here.

One: we want to prevent the system from disallowing the following cards.

1. ACTION
2. ENTRY
3. PHASE
4. INCLUDE

We accomplish this by changing $JOBCTLA. Punch out IJBJC1 ($JOBCTLA) and REP at location 2ED6 with a 47FØ.

Two: we want to make all assignments in Foreground valid. Punch out IJBJC2 ($JOBCTLD) and REP at location 2CD6 with a 47FØ.

Three: We want to make all "option" parameters valid for Foreground. Punch out IJBJC3 ($JOBCTLG) and REP at location 3D3A with a 07F7.

We must make sure the work files for the compilers and the Linkage Editor be assigned to disk. For the Linkage Editor, punch out IJBLE1 ($LINKEDT) and REP at location ØFBØ with a 47FØ. This assigns work files to a 2311. For a 2314, REP at location ØFBE with a 47FØ.

For COBOL, punch out IJSCBLØ1 and REP at location 5A with a 47FØ. Make the same change for a 2314.

For RPG, punch out IJRØØØ and REP at location D5Ø with a 47FØ,864C for 2311's. If you have 2314's REP at Location D5Ø with a 47FØ,865C.

When these changes are made, catalog IJBSL2 to F2 or F1 with this job stream:

```
// JOB CATAL
// OPTION CATAL
   ACTION F2 (or F1)
   INCLUDE IJBLE
// EXEC LNKEDT
// OPTION CATAL
   ACTION F2 (or F1)
   INCLUD IJBSL2
// EXEC LNKEDT
/&
```

The final step is to Linkedit the compilers from Foreground.

The compilers will run in Background without re-linkediting, but core must not be allocated to Foreground. All maintenance functions, except condense of core image library, will run in Foreground. Condense of core image library will run in Background, but core must not be allocated to foreground. The assembler (IJQD32) and 44K assembler (IJYASM) do not need modifications to run in Foreground. Changes to PL/I are similar to COBOL.

With these changes, it is impossible to destroy system directories by updating the directories from two partitions. The reason being that Linkedt will run only in one partiton at a time.

DOS/360 RELEASE 20 TRACE MACRO

The coding for a DOS Release 20 ALP TRACE Macro is similar in operation to the COBOL READY TRACE statement. In Exhibit 1 the Macro provides a label and subroutine trace on SYSLST. It remains a part of the program and is activated by setting UPSI switch X 01 on.

The user includes the Macro definition coding directly following the // EXEC ASSEMBLY statement and then codes the TRACE Macro at any point in his program at which a trace is desired.

Example:

```
// EXEC ASSEMBLY
       Macro
         Definition
         Coding
         .
         .
         .
START      BALR 5,0
           USING *,5
RTNA       TRACE
             .
             .
             .
RTNB       TRACE
             .
             .
             .
           END  START
```

As the labels RTNA and RTNB are reached the names 'RTNA' and 'RTNB' are printed on syslst.

DOS/360 RELEASE 20 TRACE MACRO

```
STMT   SOURCE STATEMENT

   1           MACRO
   2 &NAME     TRACE
   3           GBLB  &B1
   4 &NAME     BCR   0,0
   5           AIF   (&B1 EQ 1).T1
   6 &B1       SETB  (1)
   7           B     T&SYSNDX
   8 TRACE     MVC   TMSG,0(1)
   9           COMRG
  10           TM    23(1),X'01'
  11           BZ    X&SYSNDX
  12           EXCP  CCB&SYSNDX
  13           WAIT  CCB&SYSNDX
  14 X&SYSNDX  XC    TMSG,TMSG
  15           BR    14
  16 CCB&SYSNDX CCB  SYSLST,CCW&SYSNDX
  17 CCW&SYSNDX CCW  9,TMSG,X'20',8
  18 TRCSV     DS    4F
  19 TMSG      DS    D
  20 .T1       ANOP
  21 T&SYSNDX  STM   14,1,TRCSV
  22           L     15,=A(TRACE)
  23           LA    1,=CL8'&NAME'
  24           BALR  14,15
  25           LM    14,1,TRCSV
  26           MEND
  27           MACRO
  28 &NAME     READK &FILE,&ERR
  29 &NAME     READ  &FILE,KEY
  30           WAITF &FILE
  31           L     15,=A(ISIOCK)
  32           L     1,=A(&FILE)
  33           BALR  14,15
  34           C     1,=F'5'
  35           BL    &ERR
  36           MEND
```

EXHIBIT 1

DOS/360 LABEL PROCESSING AND FILE DESCRIPTION

Problem: Incomplete file processing

The User must supply a DLBL for each logical file to be processed and an EXTENT for each area on the DASD device. OPEN uses the information supplied by these statements and also certain information from the DTF table.

For input, the extent(s) for a file must either coincide with, or be within, an existing extent(s) that is defined in the VTOC. IOCS opens only an existing file or a subset of an existing file.

END-of-FILE is determined by either the last extent specified or by an end-of-file record on the DASD.

Normal END-of-FILE processing takes place.

Refer: Supervisor and I/O Macros
C-24-5037-5 page 25
DASD STANDARD LABELS

DOS/360 ROUTINE TO CALL SORT FROM COBOL D

This field contribution has not been submitted to any formal IBM test. Potential users should evaluate its usefulness in their own environment prior to implementation.

Using an assembly language subroutine, it is possible to gain access to the sorting capability of Sort 483 during the execution of a program written in COBOL D.

Exhibit 1 shows the COBOL statements necessary to exit to the assembly language program which calls the root phase of the sort program. 'ERROR-BYTE' is a one position field in the COBOL program which will contain a '1' if an error has occurred during sorting. The other parameters contain information which is used by the sort program.

The assembly language program, shown in Exhibit 2, first saves the base registers from the calling COBOL program. The sort parameters, passed from the COBOL program, are then initialized for the sort program. Next the address at which the sort is to be loaded is placed in register zero. This address must be on a double-word boundary above the problem program area. A code, which indicates

the successful completion of the sort program, is placed in 'RETURN'. A code of zero indicates successful completion of the sort.

The routine was tested on a 64K Model 30 using DOS Release 19.

Note that address constants of zero are set up for parameters not required or where default parameters are used. Also, there will be a possible degradation in the sort time due to loss of core depending on the size of the COBOL program.

DOS/360 ROUTINE TO CALL SORT FROM COBOL D

```
LINE NO.  SEQ. NO.          SOURCE STATEMENT

     217   110030          ENTER LINKAGE.
     218   110040          CALL 'GETDATE' USING HEAD-DATE.
     219   110050          ENTER COBOL.
     220   110060      ENTER LINKAGE.
     221   110061      CALL 'GETUPSI' USING UPSI-BYTE.
     222   110062      ENTER COBOL.
     223   110063      IF UPSI-BYTE = '10000000' GO TO ENTER-LINKAGE.
     224   110064      IF UPSI-BYTE = '01000000' GO TO START-PART-2.
   S 225   110020 START.  OPEN INPUT INV LOT OUTPUT OUT-FILE.
     226   110070 READ-INV-MSTR.
     227   110080          READ INV AT END GO TO END-PART-1.
     228   110090          IF INV-LOT-NO = 99999 GO TO END-PART-1.
     229   110100          MOVE 0 TO INDEX.
     230   110110          IF INV-IS-DELETED GO TO READ-INV-MSTR.
     231   110120          MOVE SPACES TO SORT-FILE.
     232   110130          MOVE INV-LOT-NO TO SYM-KEY S-LOT.
     233   110140 TEST-ROLL.
     234   110155          ADD 1 TO INDEX.
   S 235   110150          IF INDEX = 19 GO TO READ-INV-MSTR.
     236   110160          IF INV-ROLL-NO (INDEX) = SPACES
     237   110170                  GO TO TEST-ROLL.
     238   110175          IF LOT-KEY = SYM-KEY NEXT SENTENCE ELSE
     239   110180          READ LOT INVALID KEY MOVE SPACES TO LOT-HISTORY
     240   110190          MOVE SYM-KEY TO LOT-KEY MOVE ZEROS TO TYPE WIDTH.
     241   120020          MOVE INV-LOT-NO TO S-LOT.
     242   120030          MOVE WIDTH TO S-WIDTH.
     243   120040          MOVE TYPE TO S-TYPE.
     244   120050          MOVE INV-ROLL-NO (INDEX) TO S-ROLL.
     245   120060          MOVE INV-YARDS-PER-ROLL (INDEX) TO S-YARDS.
     246   120070          MOVE DATE-LOT TO S-DATE.
     247   120080          MOVE SUPPLIER TO S-SUPPL.
     248   120090          MOVE L-FIBER TO S-FIBER.
     249   120100          MOVE L-PRINT TO S-PRINT.
     250   120110          MOVE L-COLOR TO S-COLOR.
     251   120120          MOVE STYLE TO S-STYLE.
     252   120130          WRITE SORT-FILE.
     253   120140          GO TO TEST-ROLL.
     254   120150 END-PART-1.
     255   120160          CLOSE INV LOT OUT-FILE.
     256   120165 ENTER-LINKAGE.
     257   120170          ENTER LINKAGE.
     258   120180         CALL 'BMC$SORT' USING ERROR-BYTE  SORT-CARD  RCD-CARD
     259   120190          INPF-CARD  OUTF-CARD.
     260   120200          ENTER COBOL.
     261   120210      IF ERROR-BYTE = '1'
     262   120220      DISPLAY 'SORT ERROR -RESTART JOB'
     263   120230      STOP RUN.
     264   130010 START-PART-2.
     265   130020          OPEN INPUT OUT-FILE OUTPUT RPT.
     266   130030          PERFORM HEADING-RTN.
     267   130040 READ-SORT-FILE.
     268   130050          READ OUT-FILE AT END GO TO EOJ.
     269   130060          ON 1 MOVE S-KEY TO HOLD-KEY.
     270   130065 FIN-TOT.
```

EXHIBIT 1

DOS/360 ROUTINE TO CALL SORT FROM COBOL D

```
STMT   SOURCE STATEMENT      DOS CL3-2 09708769
  1 BEGIN    START 0
  2          ENTRY BMC$SORT
  3          USING *,5
  4 BMC$SORT SAVE  (14,12)
  5+* 360N-CL-453 SAVE     CHANGE LEVEL 3-0
  6+BMC$SORT STM   14,12,12+4*(14+2-(14+2)/16*16)(13)
  7          LR    5,15
  8          ST    13,R13
  9          L     2,0(1)       ADDRESS OF ERROR BYTE
 10          L     3,4(1)       ADDRESS OF SORT CARD
 11          L     4,8(1)       ADDRESS OF RECORD & LENGTH CARD
 12          L     6,12(1)      ADDRESS OF INPUT CARD
 13          L     7,16(1)      ADDRESS OF OUTPUT CARD
 14          MVC   SORT,0(3)    MOVE SORT CARD
 15          MVC   RCD,0(4)     MOVE RECORD & LENGTH CARD
 16          MVC   INPFL,0(6)   MOVE INPUT SIZE CARD
 17          MVC   OUTFL,0(7)   MOVE OUTPUT SIZE CARD
 18          COMRG
 19+* 360N-CL-453 COMRG     CHANGE LEVEL 3-0
 20+         L     1,20
 21+         SVC   33
 22          L     0,36(1)      LOAD DOUBLE WORD BOUNDARY
 23          C     0,BINF       ABOVE PROBLEM PROGRAM
 24          A     0,BINONE     AREA INTO R0
 25          LA    1,=CL8'SORT' LOAD ADDRESS OF SORT INTO R1
 26          SVC   4
 27          LR    15,1
 28          LA    1,PARAM
 29          LA    13,SAVAREA
 30          MVC   0(1,2),SPACE INITIALIZE ERROR-BYTE IN COBOL PROGRAM
 31          BALR  14,15        BRANCH TO SORT PROGRAM
 32          L     13,R13
 33          CLC   RETURN(2),=H'0'    TEST FOR SORT ERROR
 34          BE    RETURN2
 35          MVC   0(1,2),ONE   SET ERROR INDICATOR IN COBOL PROGRAM
 36 RETURN2  RETURN (14,12)
 37+* 360N-CL-453 RETURN   CHANGE LEVEL 3-0
 38+RETURN2  LM    14,12,12+4*(14+2-(14+2)/16*16)(13)
 39+         BR    14

 40 PARAM    DC    A(SORT)
 41          DC    A(RCD)
 42          DC    A(INPFL)
 43          DC    A(OUTFL)
 44          DC    A(0)
 45          DC    A(0)
 46          DC    A(0)
 47          DC    A(0)
 48          DC    A(0)
 49          DC    A(RETURN)
 50 SORT     DC    CL140' '
 51 RCD      DC    CL72' '
 52 INPFL    DC    CL72' '
 53 OUTFL    DC    CL72' '
 54 SAVAREA  DS    9D
 55 RETURN   DC    H'0'
 56 R13      DS    F'0'
 57 BINF     DC    BL4'1111'
 58 BINONE   DC    BL4'1'
 59 ONE      DC    CL1'1'
 60 SPACE    DC    CL1' '
 61          LTORG
 62                =CL8'SORT'
 63                =H'0'
 64          END
```

EXHIBIT 2

DOS/360 MULTIPROGRAMMING 1052 CONTENTION

This field contribution has not been submitted to any formal IBM test. Potential users should evaluate its usefulness in their own environment prior to implementation.

Certain operator messages issued on the 1052 and requiring a reply can "hang up" other partitions waiting to use the typewriter. A possible solution to this problem is to issue the message and then place the partition in the wait state. When the operator is ready to respond to the message, he presses the console interrupt key, which gives control to a STXIT OC routine. While one partition is "waiting", other partitions can be using the 1052.

This technique will work for program messages (e.g., MSG or TYPE macros) but not for DOS operator messages.

Example:

```
MSG   FORM            -request form change
WAIT  DUMMYCCB        -put partition in wait
                       state
                      -processing resumes
                       here after EXIT OC
NI    DUMMYCCB+2,     -turn off traffic bit
      Ø continue       for next wait
```

When the operator changes forms he presses the interrupt key on the console to get to a STXIT operator communication routine.

```
      STXIT            OC, RESET
                -
                -
                -

RESET OI    DUMMYCCB+2, -set on traffic bit
            X'80'
      EXIT  OC          -processing continues
                         at first instruction
                         after the wait

                -
                -
                -
DUMMYCCB    CCB   SYSLOG, DUMMYCCW,X'ØØØØ'
DUMMYCCW    CCW   X'03',*,X'2Ø',1
```

This technique has been implemented under DOS Release 19.

DOS/360 2780 FORMS JAMS

This field contribution has not been submitted to any formal IBM test. Potential users should evaluate its usefulness in their own environment prior to implementation.

There has been some confusion on how to handle forms jams and forms realignment during transmission from CPU to 2780 on a switched line. It is desirable to keep the line open for a time long enough to allow the operator to recover from printer problems but still not to tie up the line in disaster restart situations.

The 2780 sends an EOT when the printer stops because of a jam or operator intevention. Upon receiving the EOT, BTAM types '4B37I EOT RESPN' upon the console and posts a X'41' completion code with byte 24 set to X'40'. At this time the program can write ENQ sequences until the printer becomes ready or until a predetermined number of ENQ's are sent. The 2780 will answer the Write Inquiry Command with NAK's until the printer is readied. Five seconds elapse during this exchange. Therefore, for example, counting twelve Write Inquiry's will require one minute, usually time enough for the operator to recover from printer intervention situations. Receipt of NAK causes BTAM to post X'01' in byte 29 of DECB.

The coding example in Exhibit 1 shows an approach to the problem. This program reads cards and sends them unblocked to a 2780. When the EOT is received a user error counter is reset and a branch is NO-OPed to cause a return branch to the Write Initial instead of a Write Continue. As the NAK's are received they are counted. After twelve, the program writes a message to the console operator enabling him to speak to the 2780 operator at the remote location.

There are other ways to handle printer problems, but this solution has the advantage of causing only one BTAM error message to be printed on the console and provides program controlled recovery.

This program was written and tested by the contributer using DOS Release 20.

DOS/360 FORMS JAMS

```
// JOB ASSEM
// EXEC ASSEMBLY
 PRINT NOGEN
START    START 0
         BALR  10,0
         USING *,10
GO       OPEN  LIN2780,CARD
         GET CARD
WRITEI   WRITE DECB2,TI,LIN2780,FILL,84,DIAL,0,MF=E    WRITE INITIAL
         MVI   SWITCH+1,X'F0'           RESET ERROR SWITCH
         LTR   15,15                   INSURE I/O STARTED
         BNE   DUMP                    IF NOT GO TO END JOB
         BE    WAIT                     WAIT FOR COMPLETE
RETURN   GET   CARD
         WRITE DECB2,TT,LIN2780,FILL,84,MF=E   WRITE CONTINUE
TEST     LTR   15,15
         BNE   DUMP
*
*  THIS IS THE COMMON ERROR CHECKING ROUTINE
*
WAIT     WAIT  ECB=DECB2                COMMON WAIT ROUTINE
ERRCHK   CLI   DECB2,X'7F'              GOOD RETURN CODE
         BE    SWITCH                   YES
         CLI   DECB2,X'41'              IO ERROR
         BE    BITTEST                  YES
         B     DUMP                     GO TO OTHER CHECKS
BITTEST  TM    DECB2+24,X'40'           EOT RECEIVED
         BZ    NAKTEST                  NO
         MVC   CTR,=PL2'0'              YES- RESET ERROR CTR
         B     WRITINQ                  GO TO WRITE INQ
NAKTEST  TM    DECB2+29,X'01'           NAK  RECEIVED
         BZ    DUMP                     NO-GO TO OTHER CHECKS
         AP    CTR,=PL1'1'              ADD ONE TO ERROR CTR
         CP    CTR,=PL2'12'             12 TIMES (60 SEC)
         BE    CONSWT                   YES-WRITE TO OPERATOR
WRITINQ  WRITE DECB2,TQ,LIN2780,,,LIST,MF=E    WRITE INQUIRY
         LTR   15,15
         BNE   DUMP
         MVI   SWITCH+1,X'00'           SET SWITCH TO GO TO WRITE INIT
         B     WAIT                     GO TO WAIT
SWITCH   B     RETURN                  NORMALLY POINTS TO WRITE CONTINUE
*                                      BUT IS NO-OP DURING ERROR RECOVERY
         B     WRITEI  WILL GO TO WRITE INITIAL ON ERROR RECOVERY
CONSWT   PUT CONSOLE                    WRITE ERROR MESSAGE
*        FURTHER OPERATOR OPTIONS COULD BE INSERTED HERE
DUMP     DUMP                           COMMON TERMINATE ROUTINE
*        FURTHER ERROR CHECKING COULD BE DONE AT THIS POINT
CLOSE    WRITE  DECB2,TD,MF=E
       CLOSE    LIN2780,CARD
         EOJ
CTR      DC     PL2'0'                  ERROR COUNTER
LIST      DC    A(CARDIN,80)            LIST FOR WRITE INQ
LIN2780     DTFBT LINELST=(010),CU=2701,DEVICE=2780,FEATURE=(BSC,SLV), X
              MODELST=(4),CTLCHAR=EBCDIC,MODNAME=DM,CONFIG=PPT
DIAL     DFTRMLST    DIALST,0
         READ DECB2,TI,LIN2780,,,,,MF=L
CARD     DTFCD EOFADDR=CLOSE,IOAREA1=CARDIN,DEVADDR=SYS009
CONSOLE  DTFCN IOAREA1=CONSAREA,DEVADDR=SYSLOG,BLKSIZE=31
CONSAREA DC    C'PRINTER DOWN AT REMOTE LOCATION'
FILL     DC    X'0227E2'       STX  ESC  SP2
CARDIN    DC    CL80' '
         DC    X'03'
         DC    100CL4'X'
MSG     DC  C'ENTER ERR '
DM       BTMOD SWITCH=YES,BSC=YES
         END   START
/*
```

EXHIBIT 1

TYPE III AND IV PROGRAMS

Type III programs are those which have been submitted by one or more IBM employees. They are programs of general interest submitted for unrestricted distribution. They have met a basic set of programming and documentation standards but are not program tested in any formal fashion by the IBM Corporation. These programs were available for delivery from the Program Information Department prior to June 23, 1969.

Type IV programs are those contributed for unrestricted distribution by one or more authors of which at least one is an employee of an IBM customer. They are made available by IBM essentially in the author's original form, but conform to published Type IV standards. IBM exercises no control over the technical content of the documentation but merely assures that the quality of reproduction is satisfactory. Type IV programs have not been tested by IBM.

IBM makes no warranty, expressed or implied, as to the documentation, function or performance of these programs, and the user is expected to make the final evaluation as to their usefulness in his own environment.

---------- TYPE III REVISION ----------

360D-03.3.004 PL/I - FORMAC SYMBOLIC MATHEMATICS INTERPRETER. Material Revised: Documentation, Machine Readable and Catalog Abstract. Date Revised: 9/17/69.

---------- TYPE IV REVISION ----------

1401-10.3.072 FINANCIAL AID INFORMATION SYSTEM (LOAN SUBSYSTEM). Material Revised: Documentation and Machine Readable. Date Revised: 9/19/69.

---------- TYPE IV REVISION ----------

1401-10.3.073 FINANCIAL AID INFORMATION SYSTEM (STUDENT AID SUBSYSTEM). Material Revised: Documentation and Machine Readable. Date Revised: 9/19/69.

---------- NEW TYPE IV PROGRAM ----------

NON-LINEAR LEAST-SQUARES CURVE-FITTING PROGRAM

DESCRIPTION - This computer program allows the user to estimate the coefficients of a non-linear equation

$$Y = A / (x + B)^2 \quad \text{and} \quad Y = Ax^B + C$$

-- equations that are non-linear in the coefficients. An iterative technique is used; the estimates at each iteration are obtained by Marquardt's Maximum Neighborhood Method which combines the Gauss (Taylor Series) Method and the Method of Steepest Decent.

Since numerous forms of equations can be used, the user must specify the form by providing a subroutine to compute the values of the equation's coefficients. In addition, the user must provide a control card, a format card for reading data and estimates of the starting values of the coefficients. If desired, information cards and coefficient name cards can be read for display on the printout. Such displays are helpful to record the form of equation, the purpose of the run and any additional information that may help identify the printout in the future. Identification of the coefficients by name is particularly helpful when working with large or complex equations.

The output of the program is a printed report which includes a description of the problem, the starting values of the coefficients, the size of the incremental steps, a summary of each iteration and a summary of the final fit (in terms simular to those in the Linear Least-Squares Curve-Fitting Program). The statistics calculated include the number of observations, the number of coefficients, the residual degrees of freedom, the maximum and minimum value of the dependent variable

as well as its range, the standard error and t-value for each coefficient, the residual sum of squares, the residual mean square and the residual root mean square.

Listings are made of the observed and fitted values of the dependent variable--both in the sequence in which observations were given to the computer, and in the order of the magnitude of the differences between the observed and fitted values. Plots are made to indicate (1) whether these differences are normally distributed and (2) how they are distributed over all the fitted values of the dependent variable.

Provisions are made to run multiple problems as well as different equations using the same data. The program as dimensioned (256K) will handle up to 80 variables and 170 observations. Information is given in the program listings on which dimensions to change in order to reduce the overall dimensions or to increase either the number of variables or the number of observations the program will handle.

PROGRAMMING SYSTEMS - Written in OS FORTRAN IV H level.

MINIMUM SYSTEM REQUIREMENTS - A FORTRAN IV compiler, a card reader and a printer.

BASIC PROGRAM PACKAGE
DOCUMENTATION - Write-up.
MACHINE READABLE - Source code and sample problem.

OPTIONAL PROGRAM PACKAGE - None.

ORDERING INFORMATION: PROGRAM NUMBER 360D136007

	PROGRAM NUMBER EXTENSION	DISTRIBUTION MEDIUM TYPE	CODE	USER VOLUME REQUIREMENT
BASIC	none	DTR*	00	none
OPTIONAL	none	none		none

---------- NEW TYPE IV PROGRAM ----------

LINEAR LEAST-SQUARES CURVE FITTING PROGRAM

DESCRIPTION - This computer program allows the user to transform data into an appropriate form, fits specified equations to the transformed data by linear least-squares, and provides both statistics and plots to aid in evaluating the fit. A Cp-statistic search technique determines if smaller sets of the variables will represent the data equally well.

The transformations which are available to the user include reciprocals, sums, differences, products, quotients, logarithms, and exponentials. Such transformations are used to convert the observed data to more convenient or more rational units to add terms that are functions of the data-variables, to stabilize variance, and to omit variables.

In addition to the usual statistics, the program calculates the maximum and minimum value of each variable as well as its range, the relative influence of each variable and the weighted squared standardized distance of each observation from the centroid of all observations. Near neighbors are used to estimate the standard deviation of the dependent variable. A table of component effect of each variable on the fitted value of each observation shows which observations were influential in estimating the value of individual coefficients.

Listings are made of the observed and fitted values of the dependent variable--both in the sequence in which observations were given to the computer, and in the order of the magnitude of the differences between the observed and fitted values.

Plots are made to indicate (1) whether these differences are normally distributed and (2) how they are distributed over all the fitted values of the dependent variable.

The program, as dimensioned, will handle up to 80 variables and 1000 observations. Multiple dependent variables are fitted one at a time and multiple forms of specified linear equations can be fitted with one data loading.

Information is given within the program on which dimensions to change in order to reduce the overall dimensions of the program or to increase either the number of variables or the number of observations the program will handle. The current overall dimensions of the program is 256K.

PROGRAMMING SYSTEMS - Written in OS FORTRAN IV H level.

MINIMUM SYSTEM REQUIREMENTS - The machine requirements are a FORTRAN IV compiler, a card reader, four scratch files and a printer.

BASIC PROGRAM PACKAGE
DOCUMENTATION - Write-up.
MACHINE READABLE - Source code and sample problem.

OPTIONAL PROGRAM PACKAGE - None.

ORDERING INFORMATION: PROGRAM NUMBER 360D136008

	PROGRAM NUMBER EXTENSION	DISTRIBUTION MEDIUM TYPE	CODE	USER VOLUME REQUIREMENT
BASIC	none	DTR*	00	none
OPTIONAL	none	none		none

---------- NEW TYPE IV PROGRAM ----------

TAPE FILE EXTENSION MACRO - INSTRUCTION

DESCRIPTION - This macro allows the user to add records to an existing labeled tape file under DOS, eliminating the need for copying a file when making additions to it.

PROGRAMMING SYSTEMS - Written in Assembly language; operates under DOS/360.

MINIMUM SYSTEM REQUIREMENTS - Same as those required for DOS.

BASIC PROGRAM PACKAGE
DOCUMENTATION - Write-up.
MACHINE READABLE - Source code.

OPTIONAL PROGRAM PACKAGE - None.

ORDERING INFORMATION: PROGRAM NUMBER 360D003029

	PROGRAM NUMBER EXTENSION	DISTRIBUTION MEDIUM TYPE	CODE	USER VOLUME REQUIREMENT
BASIC	none	DTR*	00	none
OPTIONAL	none	none		none

---------- NEW TYPE IV PROGRAM ----------

44PS LOADER

DESCRIPTION - The 44PS Loader is a system routine designed to provide many features of the standard 44PS Linkage Editor while providing considerable performance improvements for the Model 44 FORTRAN batch processing environment. The loader can be loaded from the phase library (SDSABS) by using an EXEC card. Various execution time options are provided, including the ability to specify private relocatable libraries on both tape and disk. This permits separate libraries for each application requiring special application-oriented modules.

PROGRAMMING SYSTEMS - The program is written in Assembly Language and has been tested with 44PS Versions 3 and 4; however, the program should be release independent.

MINIMUM SYSTEM REQUIREMENTS - The required machine configuration is identical to that for 44PS.

BASIC PROGRAM PACKAGE
DOCUMENTATION - Write-up.
MACHINE READABLE - Object code, source code and sample problem.

OPTIONAL PROGRAM PACKAGE - None.

ORDERING INFORMATION: PROGRAM NUMBER 360D055001

	PROGRAM NUMBER EXTENSION	DISTRIBUTION MEDIUM TYPE	CODE	USER VOLUME REQUIREMENT
BASIC	none	DTR 9/800	28	none
		DTR 9/1600	29	none
		DTR 7DC/800	26	none
OPTIONAL	none	none		none

---------- NEW TYPE IV PROGRAM ----------

UDAC 1130 DSORT

DESCRIPTION - UDAC 1130 DSORT is a general-purpose program to sort FORTRAN-compatibile disk files. Emphasis is placed on generality of application and speed of execution. DSORT is controlled by 3-4 sort control cards which describe files and records, sorting keys, and collating sequence.

PROGRAMMING SYSTEMS - DSORT operates under the IBM 1130 Disk Monitor System Version 2 and was written in 1130 AL.

MINIMUM SYSTEM REQUIREMENTS - Those required by the 1130 Disk Monitor System (Version 2). (DSORT can use 1-3 disk drives for data storage).

BASIC PROGRAM PACKAGE
DOCUMENTATION - Write-up.
MACHINE READABLE - Source code.

OPTIONAL PROGRAM PACKAGE - None.

ORDERING INFORMATION: PROGRAM NUMBER 1130061006

	PROGRAM NUMBER EXTENSION	DISTRIBUTION MEDIUM TYPE	CODE	USER VOLUME REQUIREMENT
BASIC	none	CARDS	15	none
OPTIONAL	none	none		none

---------- NEW TYPE IV PROGRAM ----------

PURDUE SUPERMARKET MANAGEMENT GAME, 1969

DESCRIPTION - This program simulates the market environment in which supermarkets compete for sales. From two to five teams (players) can compete directly in a market area (set). Each team makes 20 decisions which would normally be made or reviewed weekly. The game administrator can influence the market environment. The supermarket is departmentalized for produce, meat, groceries and dairy products. The game can be used to teach business planning techniques, economic and accounting principles or characteristics of the firm and the industry.

PROGRAMMING SYSTEMS - This game is written in FORTRAN IV and contains one main and six subroutines. Read and write statements have been written in variable form for easy adaptation to other computer equipment.

MINIMUM SYSTEM REQUIREMENTS - An 1130 8K Disk System with a 1442 Card Read/Punch and a 1403 Printer. Unused core is 792 (HEX) words.

BASIC PROGRAM PACKAGE
DOCUMENTATION - Write-up.
MACHINE READABLE - Source code and sample problem.

OPTIONAL PROGRAM PACKAGE - None.

ORDERING INFORMATION: PROGRAM NUMBER 1130155002

	PROGRAM NUMBER EXTENSION	DISTRIBUTION MEDIUM TYPE	CODE	USER VOLUME REQUIREMENT
BASIC	none	CARDS	15	none
OPTIONAL	none	none		none

---------- NEW TYPE IV PROGRAM ----------

SIMPLIFIED GAUSS-JORDAN METHOD OF MATRIX INVERSION

DESCRIPTION - The purpose of this FORTRAN IV - written subroutine is the calculation of the inverse of any nonsingular square matrix. This program is especially valuable in many statistical and operations research problems which require the matrix inverse. The inherent value of this routine as compared to the standard Gauss-Jordan elimination technique is its speed of execution. One limiting condition is the failure of the routine of EACH AND EVERY ELEMENT of the matrix to be inverted contains the numeric value of zero.

PROGRAMMING SYSTEMS - Written in FORTRAN IV.

MINIMUM SYSTEM REQUIREMENTS - This subroutine will function properly under any operating system which supports a FORTRAN IV compiler.

BASIC PROGRAM PACKAGE
DOCUMENTATION - Write-up.
MACHINE READABLE - Source code.

OPTIONAL PROGRAM PACKAGE - None.

ORDERING INFORMATION: PROGRAM NUMBER 1130450001

	PROGRAM NUMBER EXTENSION	DISTRIBUTION MEDIUM TYPE	CODE	USER VOLUME REQUIREMENT
BASIC	none	CARDS	15	none
OPTIONAL	none	none		none

---------- NEW TYPE IV PROGRAM ----------

FUNCTION SF FOR ANY 1500 COURSEWRITER II CAI SYSTEM

DESCRIPTION - The SF function is a coursewriter function having ten sub-functions (capabilities). It can: (0) Load the contents of a return register into a buffer; (1) Put the month and day into 2 counters; (2) Store counter values in a buffer; (3) Put the clock time in tenth seconds into a counter; (4) Initialize a range of counters; (5) Initialize a range of switches; (6) Load the contents of a buffer into a return register; (7) Generate random numbers; (8) Put the time of day in tenth minutes into a counter; (9) Move the contents of one return register into another.

PROGRAMMING SYSTEMS - Written in 1130 Assembler language; operates under 1500 O/S.

MINIMUM SYSTEM REQUIREMENTS - Those required by 1500 O/S.

BASIC PROGRAM PACKAGE
DOCUMENTATION - Write-up.
MACHINE READABLE - Object code.

OPTIONAL PROGRAM PACKAGE
DOCUMENTATION - None.
MACHINE READABLE - Source code.

ORDERING INFORMATION: PROGRAM NUMBER 1500990001

	PROGRAM NUMBER EXTENSION	DISTRIBUTION MEDIUM TYPE	CODE	USER VOLUME REQUIREMENT
BASIC	none	CARDS	15	none
OPTIONAL	none	CARDS	15	none

ACCUMULATIVE LIST OF NEW TYPE III AND IV PROGRAMS

Following is an accumulative list of new Type III and Type IV programs which have been made available since the last edition of the Catalogs of Programs or their Supplements. Abstracts describing these programs can be found in the Installation Newsletter Issue No. shown below.

Program Number	Program Title	Newsletter Issue No.
360D-00.0.017	SPOOL, A PACKAGE TO CONVERT CURRENT OR FUTURE PRINTED REPORTS TO MAGNETIC TAPE FOR FOREGROUND OR BACKGROUND PRINTING LATER	69-09
360D-00.0.020	2495 UTILITY PROGRAM FOR OS/360	69-11
360D-00.0.022	PUFER - PUNCH FEED READ UTILITY CARD TO TAPE	69-15
360D-00.1.021	S/360 GENERALIZED EXTRACTION AND MATCH PROGRAM	69-07
360D-00.1.022	NYU PRINT-UTILITY PROGRAM	69-07
360D-00.2.006	OS SYSTEM/360 FLOWCHART (OS FLOW): DOS FLOWCHART UNDER OS/360	69-03
360D-00.3.025	S/360 DOS TAPE TO PRINTER/PUNCH SPOOLING SYSTEM	69-08
360D-00.3.026	FORTRAN TAPE DISK READ WRITE SUBROUTINES	69-16
360D-00.3.027	SEVEN TRACK OCTAL TAPE DUMP	69-11
360D-00.3.028	1400 INPUT/OUTPUT STANDARD LABEL PROCESSING EXITS FOR 483 DISK & TAPE SORT	69-19
360D-00.3.029	TAPE FILE EXTENSION MACRO - INSTRUCTION	69-20
360D-00.4.015	DAP DIRECT ACCESS PATCH PROGRAM	69-09

360D-00.4.018	IBM SYSTEM/360 INDEXED SEQUENTIAL DASD TO PRINT UTILITY	69-17
360D-00.5.005	FORMATTED LISTING OF VTOC FOR OS/360	69-08
360D-00.5.006	SHAREDISK	69-11
360D-00.6.011	A HYPERTEXT EDITING SYSTEM FOR THE S/360 USING THE 2250 DISPLAY	69-18
360D-01.0.009	OS/360 DATE AND TIME SUBROUTINE (DT01) WITH CONVERSION FROM JULIAN DATE TO GREGORIAN	69-09
360D-01.1.006	COMPILE AND GO FOR THE OS 360, PL/I (F) COMPILER	69-06
360D-01.3.002	1287 CHECK DIGIT AND NUMERIC RECONSTRUCTION MACROS	69-11
360D-01.4.010	SYSGEN AIDE: DIRECTORY DUMP	69-16
360D-03.0.007	S/360 MOD 44 MULTIPROGRAMMING SYSTEM	69-08
360D-03.0.008	DOS IOCS LOGIC MODULE SOURCE STATEMENT GENERATOR	69-08
360D-03.0.010	STENO TO ENGLISH TRANSLATION	69-17
360D-03.0.012	PARS CONTROL (ACP)	69-13
360D-03.1.014	FAST ASSEMBLER - INTERPRETER FOR SYSTEM/360 ASSEMBLER LANGUAGE PROGRAMS (VERSION 1)	69-14
360D-03.2.011	OS/360 PL360 COMPILER	69-06
360D-03.2.013	BASIC COMPILER UNDER TSS/360	69-13
360D-03.2.014	THE SIMSCRIPT II PROGRAMMING LANGUAGE	69-16
360D-03.2.015	THE XPL COMPILER GENERATOR SYSTEM	69-17
360D-03.3.008	COURSE III/OS A REAL-TIME COMPUTER ASSISTED INSTRUCTIONAL SYSTEM	69-01
360D-03.3.009	1404 IOCS MACROS FOR DOS/TOS	69-09
360D-03.3.010	SNAP PROCESS OR (PROTOTYPE)	69-10
360D-03.3.011	COMIT/360	69-15
360D-03.4.026	BSAM SUBROUTINE PACKAGE FOR OS 360, VERSION I	69-06
360D-03.4.027	FORTRAN RANDOM I/O SUBROUTINE	69-03
360D-03.4.028	2260 LOCAL TERMINAL USERS SYSTEM (LOTUS)	69-06
360D-03.4.031	KNUTH PROPOSAL INPUT/OUTPUT PACKAGE (KIOP)	69-16
360D-03.6.009	OS/360 COBOL CROSS-REFERENCE AND RESEQUENCING PROGRAM	69-05
360D-03.6.010	JOB STREAM MANAGER OS/MVT	69-03
360D-03.6.012	PROGRAM FOR THE CREATION OF CROSS REFERENCE TABLES, FROM S/360 RPG SOURCE PROGRAMS, FOR INDICATORS AND FIELD NAMES	69-10
360D-03.6.013	SYSTEM MANAGEMENT ROUTINES OS/MVT	69-10
360D-03.6.017	PL/60	69-15
360D-03.6.018	NEATER: A PL/I SOURCE STATEMENT REFORMATTER	69-17
360D-03.7.025	CARRIER RETURN/LINE FEED ON THE 1052/53, MISLCRLF	69-03
360D-03.7.026	DCLC A DOS/TOS/OS ASSEMBLER MACRO TO DEFINE A LOWER-CASE CHARACTER CONSTANT - DEFINE-CONSTANT-LOWER-CASE	69-09
360D-03.7.027	EDIT MACRO FOR DOS, TOS, OR OS ASSEMBLY LANGUAGE, S/360	69-07
360D-03.7.028	DOS MACRO FOR TYPEWRITER INPUT/OUTPUT	69-10
360D-03.7.031	DOS MACROS TO DUMP ISAM FILES FROM 2311 OR 2314 TO TAPE AND RESTORE FROM TAPE TO 2311 OR 2314	69-15
360D-03.8.013	PL/I STRING FUNCTIONS	69-06
360D-03.8.015	A PLAN FOR A LINKED-DIRECT FILE, WITH OPERATING ROUTINES, IN COBOL	69-13
360D-04.0.008	DOS COBOL CROSS REFERENCER	69-04
360D-04.0.009	PCMSG - PROGRAM CHECK MESSAGE	69-09
360D-04.1.010	DEBUGGING AID FOR DOS PROGRAM TESTING	69-04
360D-04.1.011	S/360 POST MORTEM DUMP (PCP/MFT)	69-09
360D-04.2.007	PROGRAM TO AID IN OPTIMIZING COMPUTE BOUND JOBS BY INDICATING THE MAJOR PROCESSING LOOPS	69-04
360D-04.6.001	A PROGRAM FOR GRAPHIC ON-LINE DEBUGGING ON AN IBM 2250 DISPLAY	69-08
360D-05.1.016	CONVERSATIONAL REMOTE BATCH ENTRY (CRBE)	69-13
360D-05.2.010	INTERACTIVE APPLICATIONS SUPERVISOR (IAS/360)	69-06
360D-05.2.011	DOS/360 TIME ACCOUNTING SYSTEMS	69-10
360D-05.5.001	44 PS LOADER	69-20

360D-06.1.005	MARC II SORT PROGRAM	69-16
360D-06.3.006	GENERALIZED TELEPROCESSING PROGRAM FOR SYSTEM/360	69-07
360D-06.3.007	DOS BTAM FOR LEASED LINE KSR 35 TELETYPES	69-06
360D-06.3.008	AUTODIN ACCESS METHOD FOR OS/360	69-10
360D-06.5.006	UNIVAC-1108 TO IBM-360 FLOATING POINT INTERNAL CONVERTER ('CVFL08')	69-06
360D-06.5.007	7090 TO 360 FLOATING POINT INTERNAL CONVERTER ('CVFL09')	69-06
360D-06.6.003	FORTRAN CHARACTER STRING PACKAGE	69-03
360D-06.6.004	CHARACTER FILTER PL1	69-03
360D-06.7.018	BSEARCH - A RANDOM ACCESS BINARY-SEARCH TECHNIQUE FOR SEQUENTIAL FILES ON DISK OR DRUM	69-03
360D-06.7.019	KWADE-KEYWORD AS A DICTIONARY ENTRY	69-03
360D-06.7.021	AUTORAF - A SYSTEM FOR BUILDING AND USING RECORD ADDRESS FILES FOR DIRECT PROCESSING OF FILES ON SYSTEM/360	69-10
360D-06.7.024	GENERATION DATA GROUP (GDG) MODIFICATION	69-10
360D-07.6.001	2741 READ AND WRITE ROUTINE	69-06
360D-08.0.003	WRIMAT MATRIX WRITER	69-05
360D-10.0.001	PROGRAM TO CALCULATE THE AMOUNT OF CPU TIME AVAILABLE IN THE BACKGROUND WHEN RUNNING DOS MULTIPROGRAMMING	69-03
360D-10.0.002	FORTRAN ANALYSIS PROGRAM	69-04
360D-10.2.004	HIGH WATER MARK (HWM)	69-09
360D-10.2.005	CORE AND/OR RESOURCE CONFLICT (CRC) NOTIFICATIONS	69-10
360D-10.2.006	NUCLEUS INITIALIZATION PROGRAM (NIP) MODIFICATIONS	69-11
360D-11.1.023	DISK RESIDENCE FOR 1410 COMPATIBILITY MODE PROGRAMS ON THE IBM S/360 MODEL 40	69-10
360D-12.1.020	PL/I BINARY TO EBCDIC CODE KEEPING FIVE BINARY CODED COLUMNS PRINTING RECODED DATA	69-11
360D-12.2.008	CDC - 3800/3300 COBOL TO IBM - S/360 COBOL TRANSLATOR	69-09
360D-12.2.009	NCR 315 TRANSLATOR FROM NEAT TO 360 ALP	69-09
360D-13.0.004	GRAPHICS STEPWISE REGRESSION ROUTINE (BMDGR1)	69-15
360D-13.0.005	NON-LINEAR LEAST SQUARES ON GRAPHICS (BMDGR2)	69-16
360D-13.0.006	TIME SERIES SPECTRUM ESTIMATION (BMDGT1)	69-15
360D-13.0.009	HISTOGRAM SUBROUTINE	69-16
360D-13.1.005	GENERAL TAXONOMY PROGRAM	69-08
360D-13.2.003	LEAST-SQUARES ESTIMATION OF NON-LINEAR PARAMETERS - NLIN	69-09
360D-13.4.007	RDSCALED RANGE TRANSFORM	69-06
360D-13.4.008	COLLEY-TUKEY FAST FOURIER TRANSFORM -- FOUR 2 IN ASSEMBLER LANGUAGE	69-17
360D-13.6.007	NON-LINEAR LEAST-SQUARES CURVE-FITTING PROGRAM	69-20
360D-13.6.008	LINEAR LEAST-SQUARES CURVE FITTING PROGRAM	69-20
360D-13.7.001	DIALL - GENERAL LEAST SQUARES DIALLEL ANALYSIS OF VARIANCE	69-15
360D-15.4.006	ICES/360 SOURCE PROJECT-I	69-05
360D-16.0.001	UCARDS: UNION CARBIDE AUTOMATIC ROUTINE AND DESIGN SYSTEM FOR PRINTED CIRCUIT BOARDS	69-15
360D-16.2.018	BULKHEAD SHEET PILING ANALYSIS	69-06
360D-16.2.019	ICES/360 STRUDL-II	69-09
360D-16.2.020	COLUMN DESIGN - WORKING STRESS METHOD	69-09
360D-16.2.021	ICES/360 SOURCE TRANSET I	69-14
360D-16.2.022	GC-21	69-19
360D-16.2.023	ICES/360 COGO-I SOURCE	69-16
360D-16.2.024	ICES/360 ROADS*I*SOURCE	69-16
360D-16.2.025	ICES/360 TABLE-II, OBJECT AND SOURCE	69-16
360D-16.3.001	ECONOMIC EVALUATION AND SCREENING PROGRAM FOR RESEARCH PROJECTS	69-05
360D-16.4.009	LISA 360 A PROGRAM FOR LINEAR SYSTEMS ANALYSIS	69-09
360D-16.4.010	SYNTH1 (A COMPUTER PROGRAM FOR THE DESIGN OF REACTIVE NETWORKS FROM A SET OF ARBITRARY POLES AND	

	ZEROES)	69-13
360D-16.8.002	COMPUTER CALCULATION OF FREQUENCY RESPONSE FUNCTIONS FROM STEP RESPONSE DATA	69-19
360D-17.7.001	SUBROUTINE UCONIC	69-07
360D-19.5.008	MVT ACCOUNTING ROUTINE (MVTACTRT)	69-12
360D-19.7.010	PROOF OF DEPOSIT ENTRY RUN WITH TRANSIT DISTRIBUTION	69-08
360D-19.7.011	PERSONAL TRUST ACCOUNTING PACKAGE TO ASSIST USERS IN INSTALLING OFF-LINE PROCESSING	69-07
360D-23.1.003	TWO-STAGE, TWO-DIMENSIONAL TRIM PROGRAM II	69-10
360D-25.0.004	EQUIPMENT ORDER, SHIPMENT, INSTALLATION, AND DISCONTINUANCE REPORT FOR LARGE ACCOUNTS	69-08
360D-25.2.004	UNIT CONTROL SYSTEM FOR RETAILERS	69-08
360D-29.3.002	VSP SCHEDULE PRODUCTION FOR O/S 360	69-16
360D-30.1.002	360/2260 FUNCTIONAL DEMONSTRATION IN DOS F1 PARTITION - BTAM LOCAL	69-09
360D-30.2.002	SHARED HOSPITAL ACCOUNTING SYSTEM (SHAS) DEMONSTRATION PACKAGE	69-09
360D-40.4.003	MULTIPLE - PRECISION FLOATING-POINT ARITHMETIC PACKAGE	69-11
360D-42.2.001	EXPERIMENTAL PROGRAM FOR DETERMINING POLYNOMIAL ZEROS	69-06
360D-45.1.001	INVRSOL	69-03
360D-45.2.001	EIGENRS	69-03
360D-99.0.004	/360 DOS INDIVIDUALIZED HOMEWORK ASSIGNMENT AND QUIZ GENERATOR (QUIZ AND EXERCISE DEMONSTRATOR: QED)	69-05
360E-00.0.008	SIMPLIFIED USER ORIENTED REPORT - WRITING PROGRAM FOR 8K MODEL 20	69-05
360E-01.0.001	INVALID CCCP/360-20 INVALID CODE CARD CORRECTION PROGRAM	69-07
360E-01.0.002	FIELD SELECTION AND LIST	69-08
360E-01.5.001	MODEL 20 STAND-ALONE DISK COPY UTILITY	69-07
360E-03.4.002	SHORT INPUT OUTPUT SYSTEM	69-06
360E-06.5.002	JULIAN DATE PROCESSING SUBROUTINES	69-08
360E-06.6.003	CONVERTING A NUMBER TO IT'S VERBAL DESCRIPTION (CKPR)	69-16
360E-13.0.001	HOSPITAL PBAR PROGRAMS FOR A MFCM MODEL 20	69-07
360E-13.0.002	STATISTICAL COMPARISONS PROGRAM	69-07
360E-13.1.006	SINGLE DIGIT COUNT AND FREQUENCY DISTRIBUTIONS	69-12
360E-13.6.001	LINEAR REGRESSION IN TWO VARIABLES (LEAST SQUARE FIT) FOR THE MOD 20	69-06
360E-16.4.004	MINIMIZATION OF BOOLEAN EXPRESSIONS	69-16
360E-16.5.001	CALCULATE MASTER CAMS	69-11
360E-19.7.020	SAVINGS ACCOUNTING FOR THE 360 MOD 20 DISK	69-04
360E-23.1.003	HEURISTIC PAPER MACHINE TRIM PROGRAM FOR THE IBM 360/20	69-10
360E-25.2.006	DOPTIM - DISK ORDER POINT TECHNIQUE OF INVENTORY MANAGEMENT	69-18
360E-27.3.001	AMERICAN FLYERS AIRLINE AVAILABILITY REPORT	69-13
360E-28.3.013	360/20 CARD DESCRIPTIVE BILLING	69-09
360E-28.3.014	WHOLESALE AND/OR RETAIL, ACCOUNTS RECEIVABLE, AGED TRIAL BALANCE	69-06
1130-00.0.009	SFREF - FORTRAN AND ASSEMBLY LANGUAGE SUBPROGRAM, LINK, AND FILE READ/WRITE CROSS-REFERENCE TABLE BUILDING PROGRAM	69-11
1130-00.3.010	1130 PRNZ MODIFIED FOR ADDITIONAL CARRIAGE CONTROL	69-07
1130-00.4.007	1130 DISK-PACK SECURITY PROGRAM	69-16
1130-01.6.003	ISFMS - 1130 INDEXED SEQUENTIAL FILE MANAGEMENT SYSTEM ON THE 1130	69-05
1130-03.4.011	REREAD FOR FREE FORMAT CARD READING IN FORTRAN	69-10
1130-05.5.001	SECAL - A PROGRAM FOR BUILDING MULTIPLE LOCAL SUBROUTINE MODULES FOR THE IBM 1130	69-10

1130-06.1.006	UDAC 1130 DSORT	69-20
1130-06.3.001	IBM 1130 REMOTE JOB ENTRY SYSTEM	69-06
1130-06.3.002	WILLIAM AND MARY REMOTE JOB ENTRY PROGRAM - 8K DISK SYSTEM VERSION (WMRJ8)	69-11
1130-06.3.003	WILLIAM AND MARY REMOTE JOB ENTRY PROGRAM - 4K DISK AND CARD SYSTEMS VERSION	69-11
1130-06.6.014	THE 1130/1800 KEYWORD INDEX GENERATOR	69-04
1130-06.6.015	IBM 1130 FORTRAN REREAD DEVICE AND FREE FORMAT KEYWORD INPUT ROUTINES	69-06
1130-06.6.016	ALPHAMERIC DATA STATEMENT PRODUCER FOR FORTRAN	69-09
1130-06.6.017	KWIC INDEX GENERATOR	69-19
1130-06.7.003	IBM 1130 INDEX FOR DM2 SYSTEMS	69-11
1130-07.0.002	RASEC/RDSEC	69-11
1130-08.0.002	HIGH SPEED AND OVERLAP 1132 PRINTER ROUTINES FOR 1130 FORTRAN AND BAL USERS	69-16
1130-12.1.005	1130 FILE CONVERSION PROGRAM, DISK MONITOR VERSION I TO DISK MONITOR VERSION 2 (FCOPY)	69-10
1130-13.7.001	MULTIPLE RANGE TEST FOR CORRELATED AND HETEROSCEDASTIC MEANS	69-08
1130-15.5.001	PURDUE FARM SUPPLY CENTER MANAGEMENT GAME	69-19
1130-15.5.002	PURDUE SUPERMARKET MANAGEMENT GAME, 1969	69-20
1130-15.2.005	1130 BLENDING OF INITIAL FURNACE CHARGES	69-10
1130-15.4.007	W.E.L. CHART SCHEDULING PROGRAM	69-16
1130-17.3.002	GENERAL PURPOSE CHROMATOGRAPH PEAK INTEGRATION PROGRAM	69-07
1130-17.6.002	1130 RAW SCORE TEST ANALYSIS PROGRAM	69-15
1130-17.6.003	1130 TEST ITEM ANALYSIS PROGRAM	69-16
1130-17.6.004	APGRD A PROGRAM TO GRADE AND ANALYZE MULTIPLE CHOICE QUIZZES	69-18
1130-21.1.001	IBM 1130 HOSPITAL PAYROLL	69-09
1130-23.3.002	COMPUTER APPROACH TO INVENTORY REQUIREMENTS ON THE 1130 FOR SMALL ASSEMBLY ORIENTED MANUFACTURERS	69-09
1130-28.2.002	1130 PATIENT BILLING	69-08
1130-28.3.001	1130 HOSPITAL ACCOUNTS RECEIVABLE	69-08
1130-30.1.004	INDIVIDUALIZED HOMEWORK ASSIGNMENT AND QUIZ GENERATOR (QUIZ AND EXERCISE DEMONSTRATOR: Q.E.D.)	69-04
1130-30.2.003	BATTLE OF THE NUMBERS	69-12
1130-30.2.004	1130 PERPETUAL CALENDAR DEMONSTRATION PROGRAM	69-13
1130-30.2.005	MOON FLIGHT	69-19
1130-45.0.001	SIMPLIFIED GAUSS-JORDAN METHOD OF MATRIX INVERSION	69-20
1500-99.0.001	FUNCTION SF FOR ANY 1500 COURSEWRITER II CAI SYSTEM	69-20
1800-00.0.002	OBJECT TIME I/O LOGICAL NUMBER GENERATOR	69-04
1800-00.0.003	CARD UTILITY PROGRAM USING 1800-OS-010, MPX	69-10
1800-01.0.002	BOM CORE & DISK DUMP UTILITIES MODIFIED TO RUN IN A CORE LOAD AREA OF V-CORE FOR USE WITH 1132 OR 1443 LINE PRINTERS	69-11
1800-17.4.001	REDUCTION OF OCEANOGRAPHIC STATION DATA	69-09
1800-23.5.006	IBM ANALOG DATA ACQUISITION PROGRAM	69-04
1800-23.5.008	PROCESS DATA PREPARATION PROGRAM	69-10
1800-25.2.001	COMPUTER PROGRAM FOR A HOSPITAL BLOOD BANK	69-19
1401-10.3.071	1401 UNIVERSITY ADMISSIONS INFORMATION SYSTEM	69-14
1401-10.3.072	FINANCIAL AID INFORMATION SYSTEM (LOAN SUBSYSTEM)	69-14
1401-10.3.073	FINANCIAL AID INFORMATION SYSTEM (STUDENT AID SUBSYSTEM)	69-14
1620-01.6.160	LABELR - A REPORT GENERATOR	69-15
1620-02.0.066	A MODIFIED PDQ FORTRAN SYSTEM	69-16
7040-H2 3552	MULTIDIMENSIONAL OPTIMIZATION USING PATTERN SEARCH	69-11
7094-F2 3554	EIGENVALUE - EIGENVECTOR ROUTINE FOR REAL COMPLEX MATRIX	69-15

IBM Installation Newsletter

December 5, 1969

Issue No. 69-24

CONTENTS

(G means Green Section)

Distribution: Branch Office-DP Management, Salesmen, Systems Engineers, FE Managers, Program Support Customer Engineers. Regions, Districts, Education Centers, Field Systems Centers, Federal Systems Centers, FE Area Offices, DPD HQ, FED HQ, WTC.

*Requires Immediate Attention

SELECTED SHORT SUBJECTS

The purpose of Selected Short Subjects is to bring together and highlight concise, factual and timely information which will indicate that action is to be taken by the IBM representative whose accounts are affected.

1. New Telephone Number for INL

A few back-of-the-envelope calculations indicate that my telephone number has been printed in the INL no less than 650,000 times during the past year. So now, they tell me, it is time for a change.

Actually, all of the phones in the White Plains building shared by DP Technical Publications have been switched to the more modern IBM Harrison telephone system with the (914) 696-1900 exchange. The new extension is 3781. However, the new exchange is a Centrex system and you can call direct by dialing (914) 696-3781.

For tie line calls, the number is now 254-3781.

Editor

DOS/360 PL/I DEVICE INDEPENDENT SYSLST LONG FILES

This contribution has not been submitted to any formal IBM test. Potential users should evaluate its usefulness in their own environment prior to implementation. Since this technique may be release dependent, it should be used with caution in order to avoid difficulty in moving to future systems.

The following modification allows DOS PL/I users to assign SYSLST to tape, disk, or printer at execution time when printing records longer than 120 bytes. Normally, if 132 character print records are assigned to tape or disk, the carriage control character is truncated. The modification was tested on Release 20.

Step 1: Modify DTFDI in the source statement library as shown in exhibit 1.

Step 2: Assemble and catalog to the relocatable library, the coding shown in exhibit 2. PL/I generates an appendage for every DTF. This appendage controls such things as page size, current line, etc. If F(133) is specified in the ENVIRONMENT statement, then PL/I generates a device dependent DTF. The coding as shown will replace the PL/I generated DTF at linkage edit time. Please note that these appendages can be found in the PL/I PLM form number Y33-9010.

Step 3: When compiling PL/I programs, the INCLUDE PRINTR card (shown in exhibit 3) must come before the EXEC PL/I card to insure the placement of the control section named PRINTR (from the relocatable library) first on SYSLNK. Then at linkage edit time, the PRINTR control section generated by PL/I is ignored.

The obvious restriction resulting from this modification is the standard file names that all programmers would have to use in their PL/I source code.

The procedure for using STREAM files is the same except for the DTF appendage which must be modified.

```
// JOB
// EXEC MAINT
   UPDATE A.DTFDI
)REP 0135
.RCSIZE1 AIF   ('&RECSIZE' GT '0' AND
'&RECSIZE' LT '134').CALC      B4530135
)REP 0137, 0138
          133 IS ASSUMED.'     B4530137
&REC     SETA 133              B4530138
)END
/&
```

EXHIBIT 1

```
PRINTR START   0
*                                      *
*DTF APPENDAGE FOR PL/I RECORD OUTPUT*
*CAN BE FOUND IN PL/I PLM Y33-9010   *
*                                      *
       DC X'A2'          OPEN MASK
       DC AL3(TABAD)     TABLE ADDRESS
       DC X'01'          FLAG BYTE ONE
       DC AL3(0)         CHAIN ADDRESS
       DC X'4500'        FLAG BYTE TWO,
                         COMM. BYTE
       DC X'0085'        RECORD LENGTH 133
       DC X'00000000'
       DC A(PBUFF)       BUFFER ADDRESS
       DC X'0085'        REMAINING DATA
       DC H'133'         DATA LENGTH
       DC H'0'           PAGESIZE
```

```
        DC H'0'          CURRENT LINE
TABAD   DC 0D'0'
*                                      *
*END OF DTF APPENDAGE                  *
*                                      *
PRNTR   DTFDI DEVADDR=SYSLST, IOAREA1=
        PBUFF, RECSIZE=133
PBUFF   DC CL133' '
        END
```

EXHIBIT 2

```
// JOB TEST PL/I MODIFICATIONS
// OPTION LINK
   INCLUDE PRINTR
// EXEC PL/I
 TEST:  PROCEDURE OPTIONS (MAIN);
DECLARE PRINTR FILE RECORD OUTPUT
ENVIRONMENT
(MEDIUM (SYSLST,1403) F(133) CTLASA);
            .
            .
            .
END;
```

EXHIBIT 3

INFORMATION ABOUT THE NEWSLETTER

EVERY SALESMAN AND SYSTEMS ENGINEER SHOULD RECEIVE ONE COPY OF THE NEWSLETTER. Each FE Branch Manager should receive five copies for distribution to Customer Engineers.

The IBM Installation Newsletter is distributed only to IBM locations and is not available to customers. The Newsletter is separated into two parts. The first part contains information which is intended only for IBM personnel and the pages are labeled For IBM Internal Use Only. THESE PAGES, OR REPRODUCTIONS OF THEM, ARE NOT TO BE GIVEN TO CUSTOMERS. However, using your own good judgement, you may discuss pertinent information from these pages with a customer. The second part contains information which may be reproduced by the Branch Office at their discretion and given to customers. These pages are not labeled with any restrictive classification. Note that when any abstracts for Type III and Type IV programs are given to the customer, he must be informed that the disclaimers for Type III and Type IV programs as contained in the Newsletter, applies. YOUR COOPERATION IN PROPERLY HANDLING THE CONTENT OF THE IBM INSTALLATION NEWSLETTER IS NECESSARY FOR ITS CONTINUED PUBLICATION.

Changes to the quantity of the Newsletter shipped are to be made by the Administration Manager. He should update and send the DP Administration Managers' Document Control card (Electro IBM N61508) to DP Document Control, DPD HQ.

Direct inquiries concerning the contents of this Newsletter to:

W. F. Malmborg, Editor
IBM Installation Newsletter
Dept. 812, DPD HQ
112 East Post Road
White Plains, New York 10601
Telephone 914 696-3781
Tie Line 254-3781

NEW PUBLICATIONS ANNOUNCED BY PRLs

Information is extracted and condensed from the weekly PRLs (Publications Release Letters) to insure that all Salesmen and Systems Engineers are aware of new or revised Reference Sources, Marketing Publications, and Field Engineering Publications. Normally, each issue of the Newsletter will contain information extracted and condensed from two PRLs, one following the other. The information will be placed in the Newsletter in its original sequence with no rearrangement of form numbers or titles. It is not intended to replace existing information and distribution sources. You should be certain that you are aware of these sources.

Do not use the condensed Newsletter information to order publications. Most but not all (for example, some films and slide sets) of the items are available from Mechanicsburg. To receive the items you need without delay, prior to ordering check the REQ (Requisition Form) code in the "Accumulative Supplement to IBM Publications Current Price List (Z20-0100)" which is attached to the weekly Publications Release Letter (PRL) and distributed to your Administration Manager. For an explanation of the REQ codes and other publications information see the first few pages of the "IBM Publications Current Price List Z20-0100".

The IBM publications requisition (MO2-0618) or ITPS should be used for ordering items from the IBM Distribution Center, Mechanicsburg, Pa.

(IBM World Trade Corporation has its own distribution center and procedures.)

PRL #46 November 14, 1969

REFERENCE SOURCES

FORM NO.	TITLE	DISPOSITION
Z Z20-1967-0	Sales and System Guide - A Bibliography of System/3 Marketing Publications and Aids	NEW

MARKETING PUBLICATIONS

P 221-0424-2	IBM 2803 Tape Control, Model 2 Proposal Insert	NEW
S 549-0718-0	MTSC Output Line Counter	NEW
S 549-0719-0	Layout Information Third Tape MT/SC	NEW
P 570-0446-0	1403 Printer Ribbon - Three Page Foldout	NEW
GA21-9103-0	IBM System/3 Card and Disk Systems Component Reference Manual	NEW
GA27-2715-2	IBM S/360 Special Feature Description 709/7040/7044/7090/7094II Compatibility Feature for IBM Models 65 and 67	Scrap A27-2715-1
G C20-1706-1	Coursewriter III Student Text	Scrap C20-1706-0
B C20-5607-3	IBM Programming the IBM 1440 Data Processing System	NEW
G E20-0198-1	Telephone Interrupt System Planning Application Manual	Scrap E20-0198-0
G H20-0367-3	System/360 Continuous System Modeling Program User's Manual (360A-CX-16X)	UES H20-0367-2, N20-1938, N20-2039
G H20-0539-1	Graphic Analysis of Three Dimensional Data Application Description Manual Program No. 5734-XX2	Scrap H20-0539-0
G H20-0627-0	System/360 Capacity Planning Infite Loading, System/360 Capacity Planning -Finite Loading Application Description Manual	
DH20-0678-0	Graphic Analysis of Three-Dimensional Data Program Description Manual Program Number 5734-XX2	NEW
DH20-0679-0	Graphic Analysis of Three-Dimensional Data Operations Manual Program Number 5734-XX2	NEW
DH20-0680-0	Graphic Analysis of Three-Dimensional Data Terminal Operations Manual Program No. 5734-XX2	NEW
DH20-0688-0	1130 Charge Materials Allocation Processor Operations Manual Program Number 5711-P11	NEW
GH20-0697-0	Medical Information System Programs Programmer's Manual Program Number 360A-UH-08L	NEW
GH20-0710-0	CALL/360 OS FORTRAN Language Reference Manual Program Number 360A-CX-42X	NEW
GN20-2007-0	TNL to 1130 Continuous System Modeling Program Applica- Description Manual Re: H20-0209-1	NEW
GN20-2064-0	TNL to S/360 Document Processing System Program Description and Operations Manual Re; H20-0477-1	NEW
GN20-2081-0	TNL to Introduction to S/360 Model 65 Multiprocessing Re: Y20-0377-0	NEW
GN22-0349-0	TNL to IBM 50 Magnetic Data Inscriber Component Description Re; A27-2725-2	NEW
SN23-0690-0	TNL to S/360 RPG Coding Re: R29-0092	NEW
GN27-2929-0	IBM S/360 Model 65 Functional Characteristics Re: A22-6884-3	NEW
GN28-0263-0	IBM S/360 Disk Operating System USA Standard COBOL Programmer's Guide Re: C28-6398-0	NEW
GN33-1530-0	TNL to S/360 Model 20 I/O Channel OEMI Re: A33-1501-0	NEW
BR20-1089-0	7080 Data Processing System Programming Video Course Study Guide	NEW
BR20-1090-0	7080 DPS Programming Video Course Work Book	NEW
KR20-4143-0	S/360 PL/I (F) Coding Workshop Student Materials	NEW
KR20-8087-0	S/360 Model 20 PL/I Coding Workshop Education Guide	NEW
PR20-9312-0	7090/94 DPS Programming Video Tape Course Description	NEW
ZY22-6949-0	IBM System Measurement Instrument Control Panel Template, Special Installation Publication	NEW
LY20-0455-0	Graphic Analysis of Three-Dimensional Data System Manual Volume 1 Program No. 5734-XX2, Feature Code 8101	NEW
LY20-0469-0	IBM 1130 Charge Materials Allocation Processor System Manual Volume 2 Compilation Listings Program No. 5711-P11	NEW
SY20-0485-0	Service for Consultants TNL - September 69	NEW
KY23-9636-0	402/403 Workshop Video Tape - Card Listing & Addition	NEW
KY23-9637-0	402/403 Workshop Video Tape - Subtraction & Selection	NEW
KY23-9638-0	402/403 Workshop Video Tape - Group Indication & Counter Coupling	NEW
KY23-9639-0	402/403 Workshop Video Tape - Minor Totals, Total Transfer	NEW
KY23-9640-0	402/403 Workshop Video Tape - Field/Class Selection & Summary Punching	NEW
KY23-9641-0	402/403 Workshop Video Tape - Carriage Control	NEW
KY23-9648-0	S/360 Remote Access Computing System - 2741 Terminal Video Tape - Introduction to RAX Program Modifications and Storage - Reel 1	NEW
KY23-9649-0	S/360 Remote Access Computing System - 2741 Terminal Video Tape - Subprogramming & Data Files - Reel 2	NEW
KY23-9819-0	Introduction to ISAM Video Tape	NEW
KY23-9820-0	OS/DASD Track and Record Formats Video Tape	NEW
KY23-9821-0	OS Linkage Conventions Video Tape	NEW
KY23-9822-0	Introduction to Library Techniques, Reel 1, Part 1	NEW
KY23-9823-0	Introduction to Library Techniques, Reel 2, Part 2	NEW
KZ23-9800-0	Introduction to Library Techniques, Reel 3, Part 2	NEW

PRL #46 (Continued)

GY26-8027-0	IBM 1500 OS Programs PLM TNL to Y26-3728-1	NEW
LYBO-0465-0	Graphic Analysis of Three-Dimensional Data System Manual Volume 2 Program Listings Program No. 5734-XX2	NEW
ZZ77-9206-0	S/360 Model 20 Programmer's Guide for RPG	NEW
ZZ77-9208-0	An On-Line Terminal System for Library Shelf List and Circulation Control of Books and Documents	NEW
ZZ77-9209-0	Textile Order Allocation for Small Systems	NEW
ZZ77-9210-0	An Evaluation of Multiprocessor 65 and the Attached Support Processor	NEW
ZZ77-9211-0	Evolution of Data Processing Systems	NEW

FIELD ENGINEERING

FORM NO.	TITLE	DISPOSITION
H132-0003-1	2420 Model 5 Magnetic Tape Unit Illustrated Parts Catalog	NEW
H229-2169-2	APAR Label	Scrap 229-2169-1
HR23-3085-2	Confirmation of Enrollment	Scrap R23-3085-1
S R23-3151-1	2303/2841 Supplementary Course Material	UES R23-3151-0
HR25-4952-2	1407 Console Inquiry Unit Student Self Study	UES R25-4952-1
HR25-5234-0	FORTRAN IV D Administrator Guide	NEW
HR25-5427-0	2025 Data Processing System Student Guide	NEW
S R25=5450-0	Safe Driving Letter	NEW
SS29-3562-0	Supplement to Student Guide Re: R31-0159-2	NEW
HV31-0192-0	Data Recorder Self Check Feature Microfiche Card	NEW
SY25-3918-0	5496 Data Recorder Student Quiz #1	NEW
SY25-3919-0	5496 Data Recorder Student Quiz #2	NEW
SY25-3920-0	5496 Data Recorder Student Quiz #3	NEW
HY26-0704-0	IBM 1500 Instructional System FEMDM FES Re: Y26-4142-0	NEW
HY26-4000-2	IBM 2841 Storage Control, Stage 2	Scrap Y26-4000-1
HY26-5902-2	IBM 1070 FEMM	Scrap Y26-5902-1
SY27-0053-0	IBM 6430 FETOM	NEW
SY27-0054-0	IBM 6430 System FEMM	NEW
SY27-0055-0	IBM 6430 System FEMDM	NEW
HY27-1040-0	Supplement to 2848 Display Control FETOM Re: Y27-2159-1	NEW
SY27-1041-0	Supplement to 2848 Display Control FEMM Re: Y27-2161-1	NEW
SY27-1042-0	Supplement to 2848 Display Control FEMDM Re: Y27-2160-1	NEW
SY27-1045-0	Supplement to 2848 Display Control FEMM Re: Y27-2140-2	NEW
SY27-2121-3	S/360 Model 65 Processor Unit FEIM	Scrap Y27-2121-2
SY27-2250-0	2385 Processor Storage Unit FEMM, FEMDM	NEW
SY27-2266-0	Supplement to Y27-2255-1	NEW
HY33-0021-0	Supplement to 2020 Processing Unit System/360 Model 20 FETOM Re: Y26-5909-5	NEW
HY33-0023-C	Supplement to S/360 Model 20 2560 Multi-Function Card Machine Attachment Feature FETOM Re: Y26-5976-3	NEW
HY33-1050-0	Supplement to Serial I/O Channel Attachment Feature, S/360 Model 20 FETOMDM Re: Y33-1040-0	NEW
ZZ22-6798-0	Supplement to Component Circuits, SLT, SLD, ASLT, MST Re: Z22-2798-0	NEW

PRL #47 November 21, 1969

FORM NO.	TITLE	DISPOSITION

MARKETING PUBLICATIONS

G 520-1965-3	For Spoken Replies from Your Computer System	UES-520-1965-2
G 520-2251-0	Advanced Systems Science Education	NEW
G A24-3312-7	2821 Control Unit	UES-A24-3312-6, N24-0428
G C24-5072-0	IBM System/360 Disk Operating System, DASD Labels	NEW
G H19-0037-0	IBM System/360 Stock Counting Option for the Retail IMPACT Staple System, Progr. Descr. Manual	NEW
G H19-0038-0	IBM System/360 Stock Counting Option for the Retail IMPACT System, Operations Manual	NEW
G K20-0348-0	Brokerage Accounting with an IBM System/360 Model 20 (Orvis Brothers and Company)	NEW
G K20-0361-0	Classroom Use of a Computer at Hotchkiss School, Lakeville, Connecticut	NEW
G N21-0105-0	TNL, Re: IBM System/360 Component Description and Operating Procedures	NEW
G N21-5111-0	IBM System/360 Operating System: Planning for the IBM 1419 Data Management Macro Instruction and Svcs.	NEW
S N23-0689-0	ENL to S/360 Model 20 RPG Coding	NEW
G N27-2938-0	S/360 Special Feature Desc. 709/7090/7094/7094II Compatibility Feature Model 85 A27-2733-0	NEW
S R20-1096-0	1410/7010 Data Proc. System Programming Video Course	NEW
S R29-0097-04	IBM System/360 Report Program Generator Coding Illus.	UES-R29-0097-3
Z X22-6948-0	IBM Systems Measurement Instrument Control Panel Planning Form	NEW
Z X22-6950-0	IBM Systems Measurement Instrument Probe Assignment Planning Form	NEW
L Y20-0468-0	IBM 1130 Charge Material Allocation Processor (1130 CMAP) System Manual - Volume 1	NEW

PRL #47 (Continued)

G Y20-0481-0	TNL to MARVEL/360 System Manual Re: Y20-0205-0	NEW
K Y23-9804-0	A Programming Language, Reel 1, Computers in the Classroom, Quik Kourse - Part 1	NEW
K Y23-9805-0	A Programming Language, Reel 2, Quik Kourse - Part II	NEW
K Y23-9806-0	A Programming Language, Reel 3, Classroom Utilization	NEW
K Y23-9807-0	A Programming Language, Reel 4, System Demonstration	NEW
K Y23-9808-0	ALC -Architecture, Reel 1, Numbering Systems	NEW
K Y23-9809-0	ALC- Architecture, Reel 2, Assembly Process	NEW
K Y23-9810-0	ALC- Architecture, Reel3, Instruction Formats	NEW
K Y23-9811-0	ALC - Architecture, Reel 4, Data Formats	NEW
K Y23-9812-0	ALC - Workshop, Reel 1, Editing	NEW
K Y23-9813-0	ALC - Workshop, Reel 2, Assembler Listing & Instruction	NEW
K Y23-9814-0	ALC - Workshop, Reel 3, Debugging - Diagnostics	NEW
K Y23-9815-0	ALC - Workshop, Reel 4, Debugging - Interrupts	NEW
K Y23-9816-0	ALC - Workshop, Reel 5, Debugging - Dump Reading	NEW
K Y23-9817-0	ALC - Workshop, Reel 6, Preview I/O	NEW
K Y23-9818-0	ALC - Workshop, Reel 7, Timer Solution	NEW
Z Z19-0013-0	TNL to IBM System/3 Marketing Guide Z19-0010-0	NEW
Z Z20-1981-0	TNL - Field Directory re: Z20-1959-3	NEW
Z Z20-1983-0	TNL to Base Publication (Z20-1897-31)	UES- Z20-1897-31

FIELD ENGINEERING

Z 123-1021-2	2020- III Processing Unit	SCRAP-123-1021-1
S 225-3486-1	1259 Magnetic Character Reader FETOM	SCRAP-225-3486-0 & S25-0088
S 225-6492-5	1403 Printers Models 1, 2, 4, 5, 6 and 7	SCRAP-225-6492-4
G 226-3011-0	FE Announcement Brochure - IBM 2770 Data Communication Sys	NEW
S 229-2127-0	Diagnostic Section Reference Manual	NEW
S 229-2166-1	IR Code Guide IBM 50 MDI	SCRAP-229-2166
S 229-4047-0	5424 Functional Unit Code Guide (Book Form)	NEW
S R23-3024-5	2403/2803 Tape Units	UES-R23-3024-4
SR23-3176-0	Monolithic System Technology SSC	NEW
S R2S-5233-0	360 FORTRAN IV D SSC	UES
S R2S-5443-0	1419/1412 Rd/sort Attach Feature SCM	NEW
S R31-0214-3	557 Alphabetic Interpreter SSC	UES-R31-0214-2
S S29-0011-0	P.M. Series II Supplement (229-1277)	NEW
S Y22-6756-0	2085 Processor Unit Compatibility Feature	NEW
S Y2S-2565-2	1052 Adopter SQ	UES-Y2S-2565-1
S Y2S-2607-1	2401/2402/2403	UES-Y2S-2607-0 & Y2S-2716-0
S Y27-1046-0	Changes to 2848 Display Control FEMDM	NEW
S Y27-1047-0	Changes to Teleprocessing - General FE Handbook	NEW
S Y27-2258-0	2065 Processing Unit Vol 1 (Y27-2036-0)	NEW
S Y27-2260-0	2065 Processing Unit (Y27-2038-0)	NEW

PRL #48 November 28, 1969

FORM NO.	TITLE	REFERENCE SOURCES	DISPOSITION
G N20-0360-63	S/360 Newsletter Re: A22-6822-13	NONE	SCRAP-N20-0360-62

MARKETING PUBLICATIONS

G 220-2004-0	Operating System/360 - PSM (GROUP 44)	NEW
G 220-2005-0	BPS/BOS/CONV/44, ETC	NEW
S 440-1027-1	General Manufacturing Invoice	UES-440-1027-0
G H20-0293-2	System/360 Flowchart (360A-SE-22X) User's Manual	UES-H20-0293-1
G A36-0001-0	IBM 4872 Modem Model 1	NEW
G C21-7513-0	Card Programming System Operator's Guide	NEW
G H20-0559-1	System/360 Matrix Language (MATLAN) Operations Manual	SCRAP-H20-0559-0
S H20-0687-0	1130 Charge Material Allocation Processor - Program Description Manual Program Number 5711-P11	NEW
S H20-0712-0	System/360 Text Processor - EDIT/360 Program Desc. Manual	NEW
G L27-3020-1	IBM 2972 Model 8 General Banking Terminal, System Component Description	SCRAP-L27-3020-0
S N23-0691-0	ENL to 24-26 Card Punch	NEW
G N28-0257-0	IBM System/360 Operating System: USA Standard Cobol Language	NEW
S R20-1093-0	1410/7010 Data Proc. System Programming Study Guide Visual Ed.	NEW
G R20-9042-2	24-26 Card Punch	NEW
G R20-9285-1	IBM 50 Mag. Data Inscriber-For Card Punch/Verifier Operators	SCRAP-R20-9285-0
G R20-9307-1	S/3 Installation Ctrl. for Management	SCRAP-R20-9307-0
S R29-0232-6	S/360 ALC Standard & Decimal Instruction P.I.	SCRAP-R29-0232-5
S R20-0234-3	IBM System/360 Assembler Language Coding	SCRAP-R29-0234-2
G X24-6514-5	Physical Planning Template - Unit Record D.P.Equipment	SCRAP-X24-6514-4
G X28-6401-0	IBM System/360 Time Sharing System, TSS/360 Quick Guide for System Programmers	NEW

PRL #48 (Continued)

L Y20-0489-0	System/360 Text Processor - Edit/360 System Manual Volume I - Flowchart Narratives	NEW
L Y20-0490-0	System/360 Text Processor - EDIT/360 System Manual Volume II - Flowcharts	NEW
G Y21-0522-0	IBM System/3 Card System Initialization and Program Maintenance PLM	NEW
Z Z19-0016-0	TNL to Z19-0007-1	NEW
Z Z20-1905-11	Inserts (less programming)	SCRAP-Z20-1905-10
Z Z20-1906-11	Sales Manual Inserts (programming only)	SCRAP-Z20-1906-10
Z Z20-1954-0	Computer Control of the Emmission Spectrometer	NEW
Z Z77-9212-0	2250 Graphic Subroutine Package Education Guide(1130)	NEW
Z Z77-9213-0	A Line Control Program Technique for the Faster Monitor with OS-BTAM	NEW
Z Z77-9214-0	ATS/360 for Remote Job Entry	NEW
Z Z77-9215-0	Introduction to Spooling and Operating Guide for Power	NEW

FIELD ENGINEERING

FORM NO.	TITLE	DISPOSITION
S 121-0544-6	2560 Illustrated Parts Catalog	SCRAP-121-0544-5
S 123-0423-3	2701 Data Adapter Unit IPC	SCRAP-123-0423-2
S 225-3502-1	1259 FEMDM	SCRAP-225-0085, 225-3502-0
S 229-2196-1	IBM 5203 Printer	SCRAP 229-2196-0
S 229-5029-4	IR CODE GUIDE - IBM 2321	UES-229-5029-3
S R23-3091-1	360 Graphic Program Service	NEW
S R25-5436-0	Mod 25 BSA Integrated Comm Attachment SG	NEW
S R27-9616-0	1800 MPX on Line Diagnostics	NEW
S R31-0399-1	Attendance Summary	NEW
S R31-0468-0	5496 Data Recorder Self-check Feature SCM	NEW
S S25-0527-0	FES to 1403 Printer Model N1 and 3 FETOM	NEW
S S25-0529-0	FES to 1219 Reader Sorter, 1419 Magnetic Character Reader FEMM	NEW
S S33-3007-0	FES for Advanced Computing Attachment for Accounting Machines, Form 225-3457-1, -2	NEW
S Y22-6801-0	FES to S/360 Model 50, 2050 Processing Unit FEMM	NEW
S Y25-2235-0	Mod 25 BSA Integrated Comm Attachment	NEW
S Y27-2259-0	2065 Processing Unit Vol 2	NEW

December 5, 1969 **IBM INSTALLATION NEWSLETTER** Issue No. 69-24

Table of Contents

OS/360 MAIN STORAGE HIERARCHY SUPPORT NOTE ADDENDUM

The following additional information has been supplied in regard to the note on the subject in INL 69-22.

There is available a circumvention for this problem. Please contact your Programming Systems CE for details.

OS/360 FORTRAN G LITERAL CONSTANTS IN DATA AND TYPE STATEMENTS

The following information is effective in FORTRAN G with OS/360 Release 18.

The FORTRAN G compiler has been modified to handle the literal constants in DATA and Type statements as follows:

1. If a single element is specified by subscripts in a DATA statement, excess characters from that element are not placed into its successor.

2. If the literal in the DATA statement is longer than one array item, and if the array name is used without a subscript, the literal will spill into succeeding items; however, this spill takes place only at the beginning of an array. If an initializing constant other than the first is longer than the element length, there will be no spill into subsequent elements.

3. If an explicit type statement contains dimension information followed by initializing values, the dimensioned array is handled as if it were an unsubscripted, previously dimensioned array in a DATA statement.

4. There is a 1 to 1 relationship between array items and initializing values so that a constant following a literal which is spilled may replace some portion of the spilled literal. See Example 2.

5. A partially filled element is padded on the right with blanks. The rest of the array is not initialized with blanks.

6. In order to start initializing in the middle of an array, EQUIVALENCE should be used.

7. An over-sized literal constant in a DATA or type statement is diagnosed (SIZE WRN-C.C.=4).

Examples:

```
(1) DIMENSION Z(10)

    DATA Z(1), Z(2), Z(6), Z(5)/

    'ABCDEFGHIJ', 'QRS', '12345', '66666'/
              $              $       $


    01) IEY038I SIZE WRN.
    02) IEY038I SIZE WRN.
    03) IEY0381 SIZE WRN.

Z(1)=ABCD
Z(2)=QRSØ
Z(3)and Z(4) will not be initialized
Z(5)=6666
Z(6)=1234
Z(7)-Z(10) will not be initialized

(2) DIEMNSION X(10)

    DATA X/ 'ABCDEFGHIJKLMNO', 10.0, 'QRST'/
                             $
    01) IEY0381 SIZE WRN.

X(1)=ABCD
X(2)=10.0
X(3)=QRST
X(4)=MNOØ
X(5)-x(10) wil not be initialized
```

OS/360 ANS COBOL OPEN PRECAUTION

The OS ANS COBOL compiler does not process OPENs properly if two or more files are named. The second and all succeeding even numbered files are not OPENed and the program will abend. PTF #27422 is available to fix the problem. The bypass is to have separate OPENs for each file.

OS/DOS/360 PL/I RECORD CONDITION U-TYPE HANDLING

For input BUFFERED files, PL/I D Version 4 and OS PL/I (F) Version 5 raise the record condition if the length of the record variable is less than the length of the data set record. For input UNBUFFERED files requiring no hidden buffers, truncation occurs, but the record condition is not raised. For output files, both BUFFERED and UNBUFFERED, the record condition is raised if the length of the record variable is greater than the length of the data set record.

Summary

I/O	BUFFERING	ACTION
INPUT	BUFFERED	RECORD RAISED IF VARIABLE < RECORD
	UNBUFFERED (& NO HIDDEN)	TRUNCATION
OUTPUT	EITHER	RECORD RAISED IF VARIABLE > RECORD

DOS/360 SHARED MODULES AND FILES PRECAUTION

DOS provides resource protection capabilities in the form of RCB, ENQ, and DEQ macros for intra-partition (i.e., subtasking) protection, and Track Hold routines for cross-partition protection. Despite careful adherence to available documentation regarding potential "lock-out" conditions, it is possible to put a multi-partition system into a WAIT state which can only be recovered from by cancelling one or more partitions. There is no error in the DOS nucleus.

A problem program requires protected access (track hold) to a file. Since this program is one subtask in a multithreaded partition, it also requires intra-partition protection of the integrity of the DTF and I/O area, etc., during the access. Finally, since an on-line environment demands efficient operations for fast response, some additional processing is necessary to allow other subtasks to access the same file.

The logic of the problem program is:

1. ENQ resource A, associated with DTFDA file A.

2. READ file A. (DTFDA and DAMOD have both HOLD=YES and RDONLY=YES specified.)

3. WAITF for I/O to complete. (No errors.)

4. Move file A data to a (dynamically obtained) unique work area.

5. DEQ resource A.

(At this point, any other subtask in the partition may access file A through DTFDA for file A. No subtask or other partition may access the specific track, however, as it is in a "hold" status.)

6. ENQ resource B, associated with ISAM file B.

7. READ file B, (no track hold capa- - bility for ISAM).

8. WAITF file B (no errors).

9. Update file B data in I/O area for file B.

10. WRITE file B record back.

11. WAITF file B (no errors).

12. DEQ resource B.

(The problem program now seeks to modify the data being retained in the work area and to rewrite file A record.)

13. Modify file A work area data.

14. ENQ resource A for file A.

15. Move updated information to file A's I/O area.

16. WRITE file A record.

17. WAITF file A (no errors).

18, FREE file A (relinquishing track hold).

19. DEQ resource A.

20. -end-

In an online situation it is possible, and indeed likely, that the same logic may be invoked by two or more subtasks concurrently. In the test being run, the above program logic was invoked by subtasks at DOS priority levels #2 and #3 (although the specific levels are unimportant).

Subtask #2 processed as far as step 8 before subtask #3 began processing. It so happened that the file A data required by subtask #3 was on the same track as that being used by subtask #2. As a result, subtask #3 was put into the WAIT state in the middle of step 2 (at the SVC-35 in DAMOD).

When subtask #2 resumed processing (in step 8) it got as far as step 14. Because subtask #3 was already ENQ'd on resource A, subtask #2 was put into the WAIT state. Thus subtask #2 could not get to step 18 to allow subtask #3 to continue. Also, a program operating independently in another partition was put into the WAIT state when it attempted to access a third record on the same track.

In sum, the sophisticated usage of a special work area to reduce the ENQ-time over resource A, designed to enhance system performance, is unworkable as it creates a potential lock-out condition. On the other hand, the elimination of the technique (specifically the work area and steps 4, 5, 14 and 15 in the above logic) has a deleterious effect on system performance, and, furthermore,

create a situation in which both resources A & B are seized simultaneously. This in itself is a risky procedure and may result in lock-out situations as described in the SRL.

DOS/360 ASSEMBLER AND LINK EDITOR IN FOREGROUND REVISION

This is a revision to the article on this subject which appeared in INL 69-10. The contribution has not been submitted to any formal IBM test. Potential users should evaluate its usefulness in their own environment prior to implementation. See INLs 69-20 and 69-22 for other articles on this subject.

A somewhat different approach was employed in installing the Foreground Assembler, COBOL Compiler, and Link Editor on Release 20 DOS. The current system requires far less operator attention and specialized Job Control. Small intermediate phases, coded as follows, were used:

```
// JOB A9400
// EXEC ASSEMBLY
   REPRO
   PHASE ASSEMBLY, +Ø
BEGIN  START   Ø
       BALR    2,Ø
       USING   *,2
       COMRG
       CLI     47(1),X'1Ø'
       BE      LOADBG
       FETCH   F2SEMBLY
LOADBG FETCH   BGSEMBLY
       END     BEGIN
/*
/&
```

The assembler and link editor changes discussed in INLs 69-10 and 69-18 still apply, but changes to job control have been simplified as follows. Four such phases are cataloged, under the names COBOL, $LNKEDT, ASSEMBLY, and $MAINEOJ, calling the corresponding phases cataloged for the partition in operation. This approach requires only three REP cards to JOB CONTROL, these being the cards to 1: allow ACTION, ENTRY, PHASE, and INCLUDE, 2: allow SYSLNK as a foreground unit, and 3: to allow the CATAL option. The LINK option requires extensive changes and is not supported for foreground. The module IJBLBH ($MAINEOJ) tests the automatic condense limits and will swing into automatic condense in foreground unless you either:

A. Set automatic condense limits to Ø or

B. REP ØØ2F6A ØØ147FØ-this overlaps a 47ØØ.

When cataloging IJBLBH for foreground, there will be one unreferenced symbol-AUTOENT-Disregard this.

With this approach there is no need for a special set of JCL for condensing the Core Image Library (no // EXEC $MAINEOJ is now required) and no special considerations for cataloging to transient or FGP directories (again, no // EXEC $MAINEOJ), and no need for operator attention in changing the EXEC cards for foreground or background compilation or cataloging.

DOS/360 FORTRAN D AND F I/O CORE REQUIREMENTS

The following is a detailed analysis of the space required for I/O buffers and DTFs in DOS/360 FORTRAN programs.

The I/O core requirements for a FORTRAN program are determined at execution time. For both FORTRAN D and F, only one buffer is assigned, to be shared by all logical units. For FORTRAN D, this buffer is assigned on the second double word boundary following the uppermost byte of the problem program. For FORTRAN F, this buffer is assigned on the first full word boundary following the uppermost byte of the problem program.

The address of the uppermost byte of the problem program is obtained from bytes 40-43 of the communication region. When executing a program which has not been cataloged into the core image library (i.e. using a //EXEC card with a blank operand), this address is the highest HICORE address appearing on the linkage edit map. When executing a program from the core image library (i.e. using a // EXEC card with a phase name operand), this address is the highest address of the problem program, where problem program is defined as all phases in the core image library beginning with the same four characters. Note that bytes 40-43 of the communication region will not be accurate if a dollar sign ($) is used as the initial character of a phase name. It is therefore prohibited in FORTRAN program phase naming to use a dollar sign as the initial character of a phase name.

The length of the buffer for FORTRAN D is the greater of 260 bytes or the largest record size in the first DEFINE FILE statement for each logical unit specified. If the buffer address plus 260 exceeds the address of the uppermost byte of the problem program partition (bytes 32-35 of the associated communication region), message IJT236I (there is not enough main storage to assign a buffer partition) is produced. When a DEFINE FILE statement for a logical unit is processed, if the address of the uppermost byte of the problem program, plus 18, plus the record size specified by that statement, exceeds the address of the uppermost byte of the problem program partition, message IJT235I (the direct access input/output record size is too large to fit in the unused main storage) is produced. In FORTRAN D all I/O is handled by DTFCP, which is included within the problem program.

The length of the buffer for FORTRAN F is the greater of (1) the largest record size allowed by a sequential logical unit used during FORTRAN execution or (2) the largest record size required by a direct access logical unit used during FORTRAN execution (record size being the record length specified in the first DEFINE FILE statement for each logical unit).

In addition to the buffer, each logical unit used in a FORTRAN F program, except for SYSLOG, requires 232 bytes in which a DTF is built dynamically as the logical unit is opened for the first time. The DTFs are built downward, with the first DTF beginning 232 bytes before the second double word boundary before the address of the uppermost byte of the problem program partition. Each subsequent DTF begins 232 bytes before the address of the last DTF built.

In the process of opening a file for the first time, if the buffer address plus the maximum record size of any logical unit already opened or the logical unit to be opened is not less than the address of the current DTF to be built, message ILF236I (there is not enough main storage to assign a buffer area) is produced.

DOS/360 SORT 483 USER EXITS CODING EXAMPLE

This field contribution has not been submitted to any formal IBM test. Potential users should evaluate its usefulness in their own environment prior to implementation.

Exhibit 1 shows a revision of Figure 24, on page 68 of SRL C28-6676-1, Tape and Disk Sort/Merge Program (360N-SM-483). This revision provides additional coding to assure proper addressability during exit processing and following completion of the Sort/Merge program. Figure 24 is not erroneous. It may, however, not be sufficiently detailed to avoid execution problems because of coding omissions. The following revised text should help clear up some programmer questions about calling the sort and processing exits. This technique has been tested by the author.

Explanation of Exhibit 1

1. This instruction loads SORT, the initial sort/merge phase. It is loaded at LOADLOC (LOADLOC is defined by instruction #47).

2. The address of the sort/merge program's entry point is placed in register 15.

3. Register 1 is loaded with the address of the address constant parameter list. This list is defined below by instructions #9 through #18.

4. This instruction loads register 13 with the user's save area address.

5. This instruction loads register 14 with the user's return address, and gives control to the sort/merge program. After sort/merge execution, control returns to the next instruction.

6. Before returning control to the user, the sort/merge program places a code in the halfword named RETURN. This instruction tests the code for successful sort/merge completion.

7. If the return code is not zero, control goes to the user's error routine.

8. Instructions begin here to process a non-zero return code upon completion of the sort.

9, 10, 11, 12, 13, 14. The sort/merge paramter list begins with #9. These first six address constants point to sort/merge control statement images that are defined below. Note that constant #13, which usually contains the address of the OPTION statement image, contains zeros. In this situation, the sort/merge program uses default values for the OPTION statement parameters.

15, 16, 17. These constants point to the required branch tables. Since no user routines are active in Phase 2, a fullword of zeros is used in place of the Phase 2 Branch Table address.

18. This is the address of a halfword in which the sort/merge program can place a return code. The halfword is defined by instruction #25.

19, 20, 21, 22, 23. These instructions define the control statement images. Note that on the MODS statement no entrie are required for the phase name and the address/length parameters, since the user routines are pre-loaded.

24. This instruciton defines an area in which the sort/merge program can save the contents of the user's registers.

25. This halfword is set aside for the sort/merge return code.

26. The literal origin instruction for all literals generated by previous coding.

27. This instruction establishes addressability for the Phase 1 Branch Table and the processing routines that follow the table.

28. This is the first instruction in the branch tables for Phase 1, the internal sort phase. Since no user routine is provided for exit E11, the sort/merge program should never give control to this instruction. If control does come to this instruction, the user's error routine is entered. (Control would come to this instruction if the user mistakenly specified E11 on the MODS statement.) If no error routine is available, unused branch table entries can be replaced with a four-byte displacement.

29. This instruciton is a branch to the user routine at exit E15. INPUT is the label of the entry point for this routine.

30, 31. Exits E17 and E18 are not used.

32. The user routine at exit E15 follows the Phase 1 Branch Table, and the first instruction should save registers that are used in the routine.

33. This instruction will restore registers to their status upon entry into the user routine and will return control to the sort/merge program.

34. Instructions begin here to process an improper Phase 1 exit.

35. The literal origin instruction for all literals generated since the previous LTORG (instruction #26), i.e., all literals generated by the Phase 1 user routines.

36. This instruction establishes addressability for the Phase 3 Branch Table and the processing routines that follow the table.

37. This is the first instruction of the Phase 3 Branch Table; no user routine is provided for exit E31.

38, 39, 40, 41. These exits are not used.

42. A user's write-error routine is active.

43. The user routine for writer errors follows the Phase 3 Branch Table, and the first instruction should save registers that are used in the routine.

44. This instruction will restore registers to their status upon entry into the user routine and will return control to the sort/merge program.

45. Instructions begin here to process an improper Phase 3 exit.

46. The literal origin instruction for all literals generated since the previous LTORG (instruction #35), i.e., all literals generated by the Phase 3 user routines.

47. The sort/merge program will be loaded here, on a double-word boundary.

DOS/360 SORT 483 USER EXITS CODING EXAMPLE

```
         --                                                          
         --                                                          
         LOAD  SORT,LOADLOC                                        1
         LR    15,1                                                2
         LA    1,PARAM                                             3
         LA    13,SAVAREA                                          4
         BALR  14,15                                               5
* FOLLOWING STATEMENTS ARE EXECUTED UPON COMPLETION OF SORT
         CLC   RETURN(2),=H'0'                                     6
         BNE   SORTERR                                             7
         --                                                          
         --                                                          
SORTERR  --                                                        8
         --                                                          
*                                                                    
PARAM    DC    A(SORT)                                             9
         DC    A(RCD)                                             10
         DC    A(INPFL)                                           11
         DC    A(OUTFL)                                           12
         DC    A(0)                                               13
         DC    A(MOD)                                             14
         DC    A(E11)                                             15
         DC    A(0)                                               16
         DC    A(E31)                                             17
         DC    A(RETURN)                                          18
SORT     DC    C'SORT FIELDS=(10,5,CH,A),WORK=5 '                 19
RCD      DC    C'RECORD TYPE=F,LENGTH=80 '                        20
INPFL    DC    C'INPFIL EXIT '                                    21
OUTFL    DC    C'OUTFIL BLKSIZE=320,OPEN=NORWD '                  22
MOD      DC    C'MODS PH1=(,,E15),PH3=(,,E39) '                   23
SAVAREA  DS    9D                                                 24
RETURN   DC    H'0'                                               25
         LTORG                                                     26
*                                                                    
* PHASE 1 BRANCH TABLE                                               
         USING E11,15                                              27
E11      B     SORTERR1                                           28
E15      B     INPUT                                              29
E17      B     SORTERR1                                           30
E18      B     SORTERR1                                           31
* PROGRAMMER'S PHASE 1 PROCESSING ROUTINES FOLLOW                    
INPUT    SAVE  (14,12)                                            32
         --                                                          
         --                                                          
         RETURN (14,12)                                           33
SORTERR1 --                                                       34
         --                                                          
         LTORG                                                    35
*                                                                    
* PHASE 3 BRANCH TABLE                                               
         USING E31,15                                             36
E31      B     SORTERR3                                           37
E32      B     SORTERR3                                           38
E35      B     SORTERR3                                           39
E37      B     SORTERR3                                           40
E38      B     SORTERR3                                           41
E39      B     WRTERR                                             42
* PROGRAMMER'S PHASE 3 PROCESSING ROUTINES FOLLOW                    
WRTERR   SAVE  (14,12)                                            43
         --                                                          
         --                                                          
         RETURN (14,12)                                           44
SORTERR3 --                                                       45
         --                                                          
         LTORG                                                    46
LOADLOC  DC    D'0'                                               47
         END
```

Exhibit 1

TYPE III AND IV PROGRAMS

Type III programs are those which have been submitted by one or more IBM employees. They are programs of general interest submitted for unrestricted distribution. They have met a basic set of programming and documentation standards but are not program tested in any formal fashion by the IBM Corporation. These programs were available for delivery from the Program Information Department prior to June 23, 1969.

Type IV programs are those contributed for unrestricted distribution by one or more authors of which at least one is an employee of an IBM customer. They are made available by IBM essentially in the author's original form, but conform to published Type IV standards. IBM exercises no control over the technical content of the documentation but merely assures that the quality of reproduction is satisfactory. Type IV programs have not been tested by IBM.

IBM makes no warranty, expressed or implied, as to the documentation, function or performance of these programs, and the user is expected to make the final evaluation as to their usefulness in his own environment.

---------- TYPE III REVISION ----------

360D-03.4.022 CATALOGUED PROCEDURES FOR OPERATION UNDER DOS/360. Material Revised: Documentation, Machine Readable, and Catalog Abstract. Date Revised: 11/14/69.

---------- TYPE IV REVISION ----------

360D-16.2.005 ICES/360 BASIC SYSTEM AND LANGUAGE PROCESSORS. Material Revised: Documentation. Note: ICES/360 BASIC SYSTEM, program number 360D-16.2.004 has been withdrawn (by the author) and its features and capabilities have been incorporated into 360D-16.2.005. Date Revised: 11/07/69.

---------- TYPE IV REVISION ----------

7040-SO 3397 CATALOGUING AND ABSTRACTING OF PROGRAMS ON 7040/44. Material Revised: Documentation (Direct Inquiries change only) and Catalog Abstract. Date Revised: 11/18/69.

---------- TYPE IV REVISION ----------

7040-SO 3398 INITIAL MASTER FILE FOR UBC CAPR. Material Revised: Documentation (Direct Inquiries Change only) and Catalog Abstract. Date Revised: 11/18/69.

---------- NEW TYPE IV PROGRAM ----------

INDEX SEQUENTIAL ERROR BYTE CHECK ROUTINE

DESCRIPTION - The Index Sequential Error Byte Check Routine will detect all error conditions which may occur while processing against or creating an index sequential file using BAL. When an error is detected the appropriate error message is displayed on the 1052 Console.

PROGRAMMING SYSTEMS - Written in DOS AL; operates under DOS/360.

MINIMUM SYSTEM REQUIREMENTS - This package requires the minimum requirements for DOS and a 1052 Console. A macro (ISERR) is used to link into the subroutine. This macro is coded immediately after the GET, PUT, READ, WRITE or WAITF to an index sequential file. The subroutine maintains a count of the records in the overflow area of the file when sequential processing is being done and when an end-of-file condition is encountered this count is displayed on the 1052 Console in a message, and control is returned to the programmer's end-of-file routine. The messages contain the DTFname and the error condition which occurred.

BASIC PROGRAM PACKAGE
DOCUMENTATION - Write-up.
MACHINE READABLE - Object code.

OPTIONAL PROGRAM PACKAGE - None.

ORDERING INFORMATION: PROGRAM NUMBER 360D020004

	PROGRAM NUMBER EXTENSION	DISTRIBUTION MEDIUM TYPE	CODE	USER VOLUME REQUIREMENT
BASIC	none	DTR*	00	none
OPTIONAL	none	none		none

---------- NEW TYPE IV PROGRAM ----------

LPI

DESCRIPTION - LPI is a small set of subprograms for use by FORTRAN programmers to perform the basic functions of list processing. This paper describes and evaluates LPI, comparing it specifically with SLIP, a similar system.

PROGRAMMING SYSTEMS - Operates under OS/360.

MINIMUM SYSTEM REQUIREMENTS - LPI requires only sufficient hardware to compile and execute FORTRAN programs.

BASIC PROGRAM PACKAGE
DOCUMENTATION - Write-up.
MACHINE READABLE - Source code and sample problem.

OPTIONAL PROGRAM PACKAGE - None.

ORDERING INFORMATION: PROGRAM NUMBER 360D068002

	PROGRAM NUMBER EXTENSION	DISTRIBUTION MEDIUM TYPE		CODE	USER VOLUME REQUIREMENT
BASIC	none	MT	9/800	28	01
		MT	9/1600	29	01
		MT	7DC/800	26	01
OPTIONAL	none	none			none

---------- NEW TYPE IV PROGRAM ----------

RAS/1130 COURSEWRITER II REASSEMBLE PROGRAM

DESCRIPTION - RAS/1130 is a disk to disk course segment reassembling program. It compresses and makes physically sequential course segments which have had statements deleted and/or added. Its effect is generally the same as the 1500 Coursewriter Reassemble Program, but RAS/1130 is considerably faster.

PROGRAMMING SYSTEMS - Written in 1130 Assembler Language; operates under the 1130 Disk Monitor, Version 2.

MINIMUM SYSTEM REQUIREMENTS - Those required by the Version 2 Disk Monitor.

BASIC PROGRAM PACKAGE
DOCUMENTATION - Write-up.
MACHINE READABLE - Object and source code.

OPTIONAL PROGRAM PACKAGE - None.

ORDERING INFORMATION: PROGRAM NUMBER 1500016001

	PROGRAM NUMBER EXTENSION	DISTRIBUTION MEDIUM TYPE	CODE	USER VOLUME REQUIREMENT
BASIC	none	CARDS	15	none
OPTIONAL	none	none		none

---------- NEW TYPE IV PROGRAM ----------

FUNCTION EP 1500 CW II FUNCTION CAI SYSTEM

DESCRIPTION - When this function is used, row and column placement of the cursor, latency, and length of response can be specified by the contents of counters (as well as constants) as in the normal EP OP code. The EP Identifier can be specified as a constant value or the contents of Buffer 5. A patch to the 1500 Operating System Interpreter is required. Use of this function on subsequent versions or levels of the operating system requires review of the patch to the Interpreter and four links to Interpreter within the function. These are covered in detail in the documentation.

PROGRAMMING SYSTEMS - This function operates under Version 3, Modification Level 1 of the 1500 Operating System. The function is used in place of the Coursewriter II OP code, EP. Source Language: 1130 Assembler.

MINIMUM SYSTEM REQUIREMENTS - 1500 system configuration with 1510 and/or 1518 Terminal Devices.

BASIC PROGRAM PACKAGE
DOCUMENTATION - Write-up.
MACHINE READABLE - Object code.

OPTIONAL PROGRAM PACKAGE - None.

ORDERING INFORMATION: PROGRAM NUMBER 1500990002

	PROGRAM NUMBER EXTENSION	DISTRIBUTION MEDIUM TYPE	CODE	USER VOLUME REQUIREMENT
BASIC	none	CARDS	15	none
OPTIONAL	none	none		none

ACCUMULATIVE LIST OF NEW TYPE III AND IV PROGRAMS

Following is an accumulative list of new Type III and Type IV programs which have been made available since the last edition of the Catalogs of Programs or their Supplements. Abstracts describing these programs can be found in the Installation Newsletter Issue No. shown below.

Program Number	Program Title	Newsletter Issue No.
360D-00.0.017	SPOOL, A PACKAGE TO CONVERT CURRENT OR FUTURE PRINTED REPORTS TO MAGNETIC TAPE FOR FOREGROUND OR BACKGROUND PRINTING LATER	69-09
360D-00.0.020	2495 UTILITY PROGRAM FOR OS/360	69-11
360D-00.0.022	PUFER - PUNCH FEED READ UTILITY CARD TO TAPE	69-15
360D-00.1.021	S/360 GENERALIZED EXTRACTION AND MATCH PROGRAM	69-07
360D-00.1.022	NYU PRINT-UTILITY PROGRAM	69-07
360D-02.0.004	INDEX SEQUENTIAL ERROR BYTE CHECK ROUTINE	69-24
360D-00.2.006	OS SYSTEM/360 FLOWCHART (OS FLOW): DOS FLOWCHART UNDER OS/360	69-03
360D-00.3.025	S/360 DOS TAPE TO PRINTER/PUNCH SPOOLING SYSTEM	69-08
360D-00.3.026	FORTRAN TAPE DISK READ WRITE SUBROUTINES	69-16
360D-00.3.027	SEVEN TRACK OCTAL TAPE DUMP	69-11
360D-00.3.028	1400 INPUT/OUTPUT STANDARD LABEL PROCESSING EXITS FOR 483 DISK & TAPE SORT	69-19
360D-00.3.029	TAPE FILE EXTENSION MACRO - INSTRUCTION	69-20
360D-00.4.015	DAP DIRECT ACCESS PATCH PROGRAM	69-09
360D-00.4.018	IBM SYSTEM/360 INDEXED SEQUENTIAL DASD TO PRINT UTILITY	69-17
360D-00.5.005	FORMATTED LISTING OF VTOC FOR OS/360	69-08
360D-00.5.006	SHAREDISK	69-11
360D-00.6.011	A HYPERTEXT EDITING SYSTEM FOR THE S/360 USING THE 2250 DISPLAY	69-18
360D-01.0.009	OS/360 DATE AND TIME SUBROUTINE (DT01) WITH CONVERSION FROM JULIAN DATE TO GREGORIAN	69-09
360D-01.1.006	COMPILE AND GO FOR THE OS 360, PL/I (F) COMPILER	69-06
360D-01.3.002	1287 CHECK DIGIT AND NUMERIC RECONSTRUCTION MACROS	69-11
360D-01.4.010	SYSGEN AIDE: DIRECTORY DUMP	69-16
360D-03.0.007	S/360 MOD 44 MULTIPROGRAMMING SYSTEM	69-08
360D-03.0.008	DOS IOCS LOGIC MODULE SOURCE STATEMENT GENERATOR	69-08
360D-03.0.010	STENO TO ENGLISH TRANSLATION	69-17
360D-03.0.012	PARS CONTROL (ACP)	69-13
360D-03.1.014	FAST ASSEMBLER - INTERPRETER FOR SYSTEM/360 ASSEMBLER LANGUAGE PROGRAMS (VERSION 1)	69-14
360D-03.2.011	OS/360 PL360 COMPILER	69-06
360D-03.2.013	BASIC COMPILER UNDER TSS/360	69-13
360D-03.2.014	THE SIMSCRIPT II PROGRAMMING LANGUAGE	69-16
360D-03.2.015	THE XPL COMPILER GENERATOR SYSTEM	69-17
360D-03.3.008	COURSE III/OS A REAL-TIME COMPUTER ASSISTED INSTRUCTIONAL SYSTEM	69-01
360D-03.3.009	1404 IOCS MACROS FOR DOS/TOS	69-09
360D-03.3.010	SNAP PROCESS OR (PROTOTYPE)	69-10
360D-03.3.011	COMIT/360	69-15
360D-03.4.026	BSAM SUBROUTINE PACKAGE FOR OS 360, VERSION I	69-06
360D-03.4.027	FORTRAN RANDOM I/O SUBROUTINE	69-03
360D-03.4.028	2260 LOCAL TERMINAL USERS SYSTEM (LOTUS)	69-06
360D-03.4.031	KNUTH PROPOSAL INPUT/OUTPUT PACKAGE (KIOP)	69-16
360D-03.6.009	OS/360 COBOL CROSS-REFERENCE AND RESEQUENCING PROGRAM	69-05
360D-03.6.010	JOB STREAM MANAGER OS/MVT	69-03
360D-03.6.012	PROGRAM FOR THE CREATION OF CROSS REFERENCE TABLES, FROM S/360 RPG SOURCE PROGRAMS, FOR INDICATORS AND	

	FIELD NAMES	6910
360D-03.6.013	SYSTEM MANAGEMENT ROUTINES OS/MVT	69-10
360D-03.6.017	PL/60	69-15
360D-03.6.018	NEATER: A PL/I SOURCE STATEMENT REFORMATTER	69-17
360D-03.7.025	CARRIER RETURN/LINE FEED ON THE 1052/53, MISLCRLF	69-03
360D-03.7.026	DCLC A DOS/TOS/OS ASSEMBLER MACRO TO DEFINE A LOWER-CASE CHARACTER CONSTANT - DEFINE-CONSTANT-LOWER-CASE	69-09
360D-03.7.027	EDIT MACRO FOR DOS, TOS, OR OS ASSEMBLY LANGUAGE, S/360	69-07
360D-03.7.028	DOS MACRO FOR TYPEWRITER INPUT/OUTPUT	69-10
360D-03.7.031	DOS MACROS TO DUMP ISAM FILES FROM 2311 OR 2314 TO TAPE AND RESTORE FROM TAPE TO 2311 OR 2314	69-15
360D-03.8.013	PL/I STRING FUNCTIONS	69-06
360D-03.8.015	A PLAN FOR A LINKED-DIRECT FILE, WITH OPERATING ROUTINES, IN COBOL	69-13
360D-04.0.008	DOS COBOL CROSS REFERENCER	69-04
360D-04.0.010	SIMPLIFIED INPUT-OUTPUT AND DEBUGGING MACROS FOR ASSEMBLER LANGUAGE USERS	69-22
360D-04.0.009	PCMSG - PROGRAM CHECK MESSAGE	69-09
360D-04.1.010	DEBUGGING AID FOR DOS PROGRAM TESTING	69-04
360D-04.1.011	S/360 POST MORTEM DUMP (PCP/MFT)	69-09
360D-04.2.007	PROGRAM TO AID IN OPTIMIZING COMPUTE BOUND JOBS BY INDICATING THE MAJOR PROCESSING LOOPS	69-04
360D-04.6.001	A PROGRAM FOR GRAPHIC ON-LINE DEBUGGING ON AN IBM 2250 DISPLAY	69-08
360D-05.1.016	CONVERSATIONAL REMOTE BATCH ENTRY (CRBE)	69-13
360D-05.1.018	BAYLOR EXECUTIVE SYSTEM FOR TELEPROCESSING (BEST)	69-22
360D-05.2.010	INTERACTIVE APPLICATIONS SUPERVISOR (IAS/360)	69-06
360D-05.2.011	DOS/360 TIME ACCOUNTING SYSTEMS	69-10
360D-05.5.001	44 PS LOADER	69-20
360D-06.1.005	MARC II SORT PROGRAM	69-16
360D-06.3.006	GENERALIZED TELEPROCESSING PROGRAM FOR SYSTEM/360	69-07
360D-06.3.007	DOS BTAM FOR LEASED LINE KSR 35 TELETYPES	69-06
360D-06.3.008	AUTODIN ACCESS METHOD FOR OS/360	69-10
360D-06.3.012	A HIGH SPEED BISYNCHRONOUS COMMUNICATIONS ACCESS METHOD	69-22
360D-06.5.006	UNIVAC-1108 TO IBM-360 FLOATING POINT INTERNAL CONVERTER ('CVFL08')	69-06
360D-06.5.007	7090 TO 360 FLOATING POINT INTERNAL CONVERTER ('CVFL09')	69-06
360D-06.6.003	FORTRAN CHARACTER STRING PACKAGE	69-03
360D-06.6.004	CHARACTER FILTER PL1	69-03
360D-06.7.018	BSEARCH - A RANDOM ACCESS BINARY-SEARCH TECHNIQUE FOR SEQUENTIAL FILES ON DISK OR DRUM	69-03
360D-06.7.019	KWADE-KEYWORD AS A DICTIONARY ENTRY	69-03
360D-06.7.021	AUTORAF - A SYSTEM FOR BUILDING AND USING RECORD ADDRESS FILES FOR DIRECT PROCESSING OF FILES ON SYSTEM/360	69-10
360D-06.7.024	GENERATION DATA GROUP (GDG) MODIFICATION	69-10
360D-06.8.002	LPI	69-24
360D-07.6.001	2741 READ AND WRITE ROUTINE	69-06
360D-08.0.003	WRIMAT MATRIX WRITER	69-05
360D-10.0.001	PROGRAM TO CALCULATE THE AMOUNT OF CPU TIME AVAILABLE IN THE BACKGROUND WHEN RUNNING DOS MULTIPROGRAMMING	69-03
360D-10.0.002	FORTRAN ANALYSIS PROGRAM	69-04
360D-10.2.004	HIGH WATER MARK (HWM)	69-09
360D-10.2.005	CORE AND/OR RESOURCE CONFLICT (CRC) NOTIFICATIONS	69-10
360D-10.2.006	NUCLEUS INITIALIZATION PROGRAM (NIP) MODIFICATIONS	69-11
360D-11.1.023	DISK RESIDENCE FOR 1410 COMPATIBILITY MODE PROGRAMS ON THE IBM S/360 MODEL 40	69-10
360D-12.1.020	PL/I BINARY TO EBCDIC CODE KEEPING FIVE BINARY	

	coded columns PRINTING RECODED data	69-11
360D-12.2.008	CDC - 3800/3300 COBOL TO IBM - S/360 COBOL TRANSLATOR	69-09
360D-12.2.009	NCR 315 TRANSLATOR FROM NEAT TO 360 ALP	69-09
360D-13.0.004	GRAPHICS STEPWISE REGRESSION ROUTINE (BMDGR1)	69-15
360D-13.0.005	NON-LINEAR LEAST SQUARES ON GRAPHICS (BMDGR2)	69-16
360D-13.0.006	TIME SERIES SPECTRUM ESTIMATION (BMDGT1)	69-15
360D-13.0.008	CAI DOS COURSEWRITER III CALCULATOR	69-21
360D-13.0.009	HISTOGRAM SUBROUTINE	69-16
360D-13.1.005	GENERAL TAXONOMY PROGRAM	69-08
360D-13.2.003	LEAST-SQUARES ESTIMATION OF NON-LINEAR PARAMETERS - NLIN	69-09
360D-13.4.007	RESCALED RANGE TRANSFORM	69-06
360D-13.4.008	COLLEY-TUKEY FAST FOURIER TRANSFORM -- FOUR 2 IN ASSEMBLER LANGUAGE	69-17
360D-13.6.007	NON-LINEAR LEAST-SQUARES CURVE-FITTING PROGRAM	69-20
360D-13.6.008	LINEAR LEAST-SQUARES CURVE FITTING PROGRAM	69-20
360D-13.7.001	DIALL - GENERAL LEAST SQUARES DIALLEL ANALYSIS OF VARIANCE	69-15
360D-15.0.004	OS/360 CHECKPOINT AND IPL RESTART	69-23
360D-15.4.006	ICES/360 SOURCE PROJECT-I	69-05
360D-16.0.001	UCARDS: UNION CARBIDE AUTOMATIC ROUTINE AND DESIGN SYSTEM FOR PRINTED CIRCUIT BOARDS	69-15
360D-16.2.018	BULKHEAD SHEET PILING ANALYSIS	69-06
360D-16.2.019	ICES/360 STRUDL-II	69-09
360D-16.2.020	COLUMN DESIGN - WORKING STRESS METHOD	69-09
360D-16.2.021	ICES/360 SOURCE TRANSET I	69-14
360D-16.2.022	GC-21	69-19
360D-16.2.023	ICES/360 COGO-I SOURCE	69-16
360D-16.2.024	ICES/360 ROADS I SOURCE	69-16
360D-16.2.025	ICES/360 TABLE-II, OBJECT AND SOURCE	69-16
360D-16.2.026	ICES/LEASE I	69-23
360D-16.3.001	ECONOMIC EVALUATION AND SCREENING PROGRAM FOR RESEARCH PROJECTS	69-05
360D-16.4.009	LISA 360 A PROGRAM FOR LINEAR SYSTEMS ANALYSIS	69-09
360D-16.4.010	SYNTH1 (A COMPUTER PROGRAM FOR THE DESIGN OF REACTIVE NETWORKS FROM A SET OF ARBITRARY POLES AND ZEROES)	69-13
360D-16.8.002	COMPUTER CALCULATION OF FREQUENCY RESPONSE FUNCTIONS FROM STEP RESPONSE DATA	69-19
360D-17.2.009	ON-LINE SUMX	69-23
360D-17.7.001	SUBROUTINE UCONIC	69-07
360D-19.5.008	MVT ACCOUNTING ROUTINE (MVTACTRT)	69-12
360D-19.7.010	PROOF OF DEPOSIT ENTRY RUN WITH TRANSIT DISTRIBUTION	69-08
360D-19.7.011	PERSONAL TRUST ACCOUNTING PACKAGE TO ASSIST USERS IN INSTALLING OFF-LINE PROCESSING	69-07
360D-23.1.003	TWO-STAGE, TWO-DIMENSIONAL TRIM PROGRAM II	69-10
360D-25.0.004	EQUIPMENT ORDER, SHIPMENT, INSTALLATION, AND DISCONTINUANCE REPORT FOR LARGE ACCOUNTS	69-08
360D-25.2.004	UNIT CONTROL SYSTEM FOR RETAILERS	69-08
360D-29.3.002	VSP SCHEDULE PRODUCTION FOR O/S 360	69-16
360D-30.1.002	360/2260 FUNCTIONAL DEMONSTRATION IN DOS F1 PARTITION - BTAM LOCAL	69-09
360D-30.2.002	SHARED HOSPITAL ACCOUNTING SYSTEM (SHAS) DEMONSTRATION PACKAGE	69-09
360D-40.4.003	MULTIPLE - PRECISION FLOATING-POINT ARITHMETIC PACKAGE	69-11
360D-42.2.001	EXPERIMENTAL PROGRAM FOR DETERMINING POLYNOMIAL ZEROS	69-06
360D-45.1.001	INVRSOL	69-03
360D-45.2.001	EIGENRS	69-03
360D-99.0.004	/360 DOS INDIVIDUALIZED HOMEWORK ASSIGNMENT AND QUIZ GENERATOR (QUIZ AND EXERCISE DEMONSTRATOR: QED)	69-05

360E-00.0.008	SIMPLIFIED USER ORIENTED REPORT - WRITING PROGRAM FOR 8K MPDEL 20	69-05
360E-01.0.001	INVALID CCCP/360-20 INVALID CODE CARD CORRECTION PROGRAM	69-07
360E-01.0.002	FIELD SELECTION AND LIST	69-08
360E-01.5.001	MODEL 20 STAND-ALONE DISK COPY UTILITY	69-07
360E-03.4.002	SHORT INPUT OUTPUT SYSTEM	69-06
360E-06.5.002	JULIAN DATE PROCESSING SUBROUTINES	69-08
360E-06.6.003	CONVERTING A NUMBER TO IT'S VERBAL DESCRIPTION (CKPR)	69-16
360E-13.0.001	HOSPITAL PBAR PROGRAMS FOR A MFCM MODEL 20	69-07
360E-13.0.002	STATISTICAL COMPARISONS PROGRAM	69-07
360E-13.1.006	SINGLE DIGIT COUNT AND FREQUENCY DISTRIBUTIONS	69-12
360E-13.6.001	LINEAR REGRESSION IN TWO VARIABLES (LEAST SQUARE FIT) FOR THE MOD 20	69-06
360E-16.4.004	MINIMIZATION OF BOOLEAN EXPRESSIONS	69-16
360E-16.5.001	CALCULATE MASTER CAMS	69-11
360E-19.7.020	SAVINGS ACCOUNTING FOR THE 360 MOD 20 DISK	69-04
360E-23.1.003	HEURISTIC PAPER MACHINE TRIM PROGRAM FOR THE IBM 360/20	69-10
360E-25.2.006	DOPTIM - DISK ORDER POINT TECHNIQUE OF INVENTORY MANAGEMENT	69-18
360E-27.3.001	AMERICAN FLYERS AIRLINE AVAILABILITY REPORT	69-13
360E-28.3.013	360/20 CARD DESCRIPTIVE BILLING	69-09
360E-28.3.014	WHOLESALE AND/OR RETAIL, ACCOUNTS RECEIVABLE, AGED TRIAL BALANCE	69-06
1130-00.0.009	SFREF - FORTRAN AND ASSEMBLY LANGUAGE SUBPROGRAM, LINK, AND FILE READ/WRITE CROSS-REFERENCE TABLE BUILDING PROGRAM	69-11
1130-00.3.010	1130 PRNZ MODIFIED FOR ADDITIONAL CARRIAGE CONTROL	69-07
1130-00.4.007	1130 DISK-PACK SECURITY PROGRAM	69-16
1130-01.5.006	DUMP AND RESTORE MONITOR VERSION TWO	69-21
1130-01.6.003	ISFMS - 1130 INDEXED SEQUENTIAL FILE MANAGEMENT SYSTEM ON THE 1130	69-05
1130-03.4.011	REREAD FOR FREE FORMAT CARD READING IN FORTRAN	69-10
1130-05.5.001	SECAL - A PROGRAM FOR BUILDING MULTIPLE LOCAL SUBROUTINE MODULES FOR THE IBM 1130	69-10
1130-06.1.006	UDAC 1130 DSORT	69-20
1130-06.1.007	ORDER. AN IN-CORE TAG SORT	69-23
1130-06.3.001	IBM 1130 REMOTE JOB ENTRY SYSTEM	69-06
1130-06.3.002	WILLIAM AND MARY REMOTE JOB ENTRY PROGRAM - 8K DISK SYSTEM VERSION (WMRJ8)	69-11
1130-06.3.003	WILLIAM AND MARY REMOTE JOB ENTRY PROGRAM - 4K DISK AND CARD SYSTEMS VERSION	69-11
1130-06.6.014	THE 1130/1800 KEYWORD INDEX GENERATOR	69-04
1130-06.6.015	IBM 1130 FORTRAN REREAD DEVICE AND FREE FORMAT KEYWORD INPUT ROUTINES	69-06
1130-06.6.016	ALPHAMERIC DATA STATEMENT PRODUCER FOR FORTRAN	69-09
1130-06.6.017	KWIC INDEX GENERATOR	69-19
1130-06.7.003	IBM 1130 INDEX FOR DM2 SYSTEMS	69-11
1130-07.0.002	RASEC/RDSEC	69-11
1130-08.0.002	HIGH SPEED AND OVERLAP 1132 PRINTER ROUTINES FOR 1130 FORTRAN AND BAL USERS	69-16
1130-12.1.005	1130 FILE CONVERSION PROGRAM, DISK MONITOR VERSION I TO DISK MONITOR VERSION 2 (FCOPY)	69-10
1130-13.7.001	MULTIPLE RANGE TEST FOR CORRELATED AND HETEROSCEDASTIC MEANS	69-08
1130-15.1.002	GASP II (GENERAL ACTIVITY SIMULATION PROGRAM, VERSION II)	69-23
1130-15.5.001	PURDUE FARM SUPPLY CENTER MANAGEMENT GAME	69-19
1130-15.5.002	PURDUE SUPERMARKET MANAGEMENT GAME, 1969	69-20
1130-15.2.005	1130 BLENDING OF INITIAL FURNACE CHARGES	69-10
1130-15.4.007	W.E.L. CHART SCHEDULING PROGRAM	69-16

1130-16.5.002	MECHANICAL TOLERANCE ANALYSIS PROGRAM (MTAP)	69-22
1130-17.3.002	GENERAL PURPOSE CHROMATOGRAPH PEAK INTEGRATION PROGRAM	69-07
1130-17.6.002	1130 RAW SCORE TEST ANALYSIS PROGRAM	69-15
1130-17.6.003	1130 TEST ITEM ANALYSIS PROGRAM	69-16
1130-17.6.004	APGRD A PROGRAM TO GRADE AND ANALYZE MULTIPLE CHOICE QUIZZES	69-18
1130-21.1.001	IBM 1130 HOSPITAL PAYROLL	69-09
1130-23.3.002	COMPUTER APPROACH TO INVENTORY REQUIREMENTS ON THE 1130 FOR SMALL ASSEMBLY ORIENTED MANUFACTURERS	69-09
1130-28.2.002	1130 PATIENT BILLING	69-08
1130-28.3.001	1130 HOSPITAL ACCOUNTS RECEIVABLE	69-08
1130-30.1.004	INDIVIDUALIZED HOMEWORK ASSIGNMENT AND QUIZ GENERATOR (QUIZ AND EXERCISE DEMONSTRATOR: Q.E.D.)	69-04
1130-30.2.003	BATTLE OF THE NUMBERS	69-12
1130-30.2.004	1130 PERPETUAL CALENDAR DEMONSTRATION PROGRAM	69-13
1130-30.2.005	MOON FLIGHT	69-19
1130-45.0.001	SIMPLIFIED GAUSS-JORDAN METHOD OF MATRIX INVERSION	69-20
1500-01.6.001	RAS/1130 COURSEWRITER II REASSEMBLE PROGRAM	69-24
1500-99.0.001	FUNCTION SF FOR ANY 1500 COURSEWRITER II CAI SYSTEM	69-20
1500-99.0.002	FUNCTION EP 1500 CW II FUNCTION CAI SYSTEM	69-24
1800-00.0.002	OBJECT TIME I/O LOGICAL NUMBER GENERATOR	69-04
1800-00.0.003	CARD UTILITY PROGRAM USING 1800-OS-010, MPX	69-10
1800-01.0.002	BOM CORE & DISK DUMP UTILITIES MODIFIED TO RUN IN A CORE LOAD AREA OF V-CORE FOR USE WITH 1132 OR 1443 LINE PRINTERS	69-11
1800-17.4.001	REDUCTION OF OCEANOGRAPHIC STATION DATA	69-09
1800-23.5.006	IBM ANALOG DATA ACQUISITION PROGRAM	69-04
1800-23.5.008	PROCESS DATA PREPARATION PROGRAM	69-10
1800-25.2.001	COMPUTER PROGRAM FOR A HOSPITAL BLOOD BANK	69-19
1401-10.3.071	1401 UNIVERSITY ADMISSIONS INFORMATION SYSTEM	69-14
1401-10.3.072	FINANCIAL AID INFORMATION SYSTEM (LOAN SUBSYSTEM)	69-14
1401-10.3.073	FINANCIAL AID INFORMATION SYSTEM (STUDENT AID SUBSYSTEM)	69-14
1620-01.6.160	LABELR - A REPORT GENERATOR	69-15
1620-02.0.066	A MODIFIED PDQ FORTRAN SYSTEM	69-16
7040-H2 3552	MULTIDIMENSIONAL OPTIMIZATION USING PATTERN SEARCH	69-11
7094-F2 3554	EIGENVALUE - EIGENVECTOR ROUTINE FOR REAL COMPLEX MATRIX	69-15

DELETED TYPE III PROGRAM

The following Type III Program will be deleted by the Program Information Department on January 5, 1970. Until that date, orders for this program will be accepted by PID. This deletion has been requested by the author.

360D-30.0.001	MACON - S/360 MANUFACTURING DEMONSTRATION PROGRAM

IBM Installation Newsletter

December 19, 1969

Issue No. 69-25

CONTENTS

(G means Green Section)

NOTE: This is the final Newsletter for 1969.
The next issue will be published January 16, 1970.
HAPPY NEW YEAR!

Distribution: Branch Office-DP Management, Salesmen, Systems Engineers, PE Managers, Program Support Customer Engineers. Regions, Districts, Education Centers, Field Systems Centers, Federal Systems Centers, FE Area Offices, DPD HQ, FED HQ, WTC.

*Requires Immediate Attention

OS/360 RELEASE 18 RELIABILITY

The following information is reprinted from RETAIN:

Release 18 feedback has shown it to have high reliability. If an account is considering installation of Release 18, it should be encouraged to do so. An updated Release 18 is still planned for accounts that wish to defer installation.

OS/360 RELEASE PROCEDURES

The following information is provided in order to help you explain and discuss this subject with your OS/360 customers. This material should not be given to a customer directly.

The Release 18 Programming Announcement (P69-119) stated that an updated release would be available in 3 to 4 months. A release is updated with PTFs so that users can benefit from problems found and corrected during the initial field exposure. After a release has been in the field for 2 to 3 months, the PTFs that were made available through normal field engineering channels, documented in RETAIN and the PSMs, are reviewed to determine which should be included in the update. The update is then created, tested and shipped to PID to replace the initial level of the release.

First results indicate that Release 18 is more reliable than prior releases. The decision on whether to install Release 18 initially or the update should be based upon individual account requirements. That is, his satisfaction with his current production release, requirements for new function, system workload, account plans and on-order equipment, etc.

There should be no reason for a user to order and install both the initial release and its update. If he installs the release initially, he can apply PTFs as they become available. The user installing the update has these PTFs applied for him centrally by IBM, rather than serially at his location.

OS releases are not fixed at 6 month intervals, but vary somewhat depending upon both internal and external schedules. In addition, there are from time to time special releases, affecting a limited number of users, which are necessary to meet our external commitments. The basic release plan is designed so that any given user of OS/360 can install 1 or 2 releases per year depending upon his requirements.

PTF support for a release of OS/360 is provided for 9 months after the follow-on release. This means a release is supported with PTFs for approximately 15 months after its initial availability.

The OS/360 release plan (reference P68-170 for details) is designed to provide users with early availability, improved reliability and maximum flexibility. The OS user may choose which release he wants to install and when he wants to install it.

OS/360 CRBE OR CRJE WITH LCS PRECAUTION

A very serious overrun problem exists when CRBE or CRJE is installed with LCS. These programs utilize data chaining with small byte counts in the channel programs for the 2314. These channel programs cannot be executed from LCS.

Implementation of Main Storage Hierarchy Support under OS/360 will enable the execution of 2314 command chains from high speed storage. Where this solution is not desirable, CRBE or CRJE must be modified to permit use in an LCS environment.

DOS/360 IPL SYSTEM IDENTIFICATION MESSAGE

This contribution has not been submitted to any formal IBM test. Potential users should evaluate its usefulness in their own environment prior to implementation.

Installations with several variations of DOS concurrently available need a simple method of having the operating system log its identification during the IPL procedure.

The 'IPLMSG' macro, Exhibit 1, when incorporated in a supervisor assembly, Exhibit 2, will produce just such a message, Exhibit 3. The code should be release independent, and can easily be extended to produce multiple lines of message.

This method of identifying the pack being IPLed has been in use at a Datacenter for about two years. Constructions of DOS on the several machines are easily distinguished.

Alternative approaches to this same problem have been suggested in INLs 68-11 and 69-19. (A method for selecting an altenate supervisor during the IPL is given in INL 68-21.*) The method given here has the advantages that it is: selectable at supervisor assembly time (use/don't use the IPLMSG card), and that it doesn't alter any IBM supplied components.

*The 68-21 article gives the core location of the '1' part of '$$A$SUP1' as X'3503' for DOS III. The correct location for DOS Releases 20 and 21 is X'351B'.

DOS/360 IPL SYSTEM IDENTIFICATION MESSAGE

```
          CATALS A.IPLMSG
          MACRO
          IPLMSG
          GBLA  &AG15
          AIF   (&AG15 GE 6).OK  MACRO SEQUENCE CHECK
          MNOTE 7,'THE REST BLEW, WHY NOT THIS TOO?'
          MEXIT
.OK       ANOP
*
*    THE FOLLOWING CODE IS PROVIDED TO GIVE SYSTEM
*    IDENTIFICATION AT IPL TIME
*
.*   IT IS INSERTED FOLLOWING THE TEMPORARY CODE IN THE LTA
.*   IN THE 'SEND' MACRO.  THE LTA IS 1200 BYTES LONG, OF
.*   WHICH APPROXIMATELY 74 IS USED BY SEND.
.*
.*   SUPERVISOR DECK ARRANGEMENT IS -
.*           - - -
.*           DVCGENS
.*           ASSGNS
.*           SEND
.*           IPLMSG
.*   ZZMSG   DC    C'MESSAGE TEXT AS DESIRED'
.*           END
.*
.*
IJBSUP20 CSECT
          USING LTA,RF .        CONTINUE BASE USED IN TEMP CODE
*
          ORG   RES2311
          BAL   R8,ZZIDCODE .   BRANCH TO PATCH CODE
*
          ORG   MVCEND .        EXTEND ONWARD INTO LTA
ZZIDCODE DS     0D
          BCTR  RE,0 .          EXECUTE OVERLAYED CODE
          BCTR  RE,0 .          *
*                               ADDED CODE STARTS HERE
          MVC   CAW,ZZCCWADR .  CCW ADDRESS TO CAW
ZZLOG     EQU   X'01F' .        1052 ADDRESS
          SIO   ZZLOG .         TYPE THE MESSAGE
          BNZ   ZZRETURN .      IF LOG DOWN, FORGET IT
          TIO   ZZLOG
          BC    2,*-4 .         LOOP WHILE TYPING
ZZRETURN BR     R8
*
ZZCCW     CCW   09,ZZMSG,X'20',L'ZZMSG
ZZCCWADR DC     A(ZZCCW)
.*
.*   THE 'ZZMSG' CARD MAY BE PUT IN RIGHT HERE, IF PREFERRED
*
          MEND
```

EXHIBIT 1

For IBM Internal Use Only

DOS/360 IPL SYSTEM IDENTIFICATION MESSAGE

```
3239+M252CB2  EQU   X'00'
3240+M1285    EQU   NBTFLG
3241+M1287    EQU   NBTFLG
3242+M2311    EQU   NBTFLG
3243+M2321    EQU   NBTFLG
3244+M2314    EQU   NBTFLG
3245+M1403    EQU   X'00'
3246+M1404    EQU   M1403
3247+M1443    EQU   X'00'
3248+M1445    EQU   M1443
3249+M1403U   EQU   X'00'
3250+M2671    EQU   NBTFLG
3251+MUNSP    EQU   X'00' 3-2
3252+MUNSPB   EQU   NBTFLG 3-2
3253          IPLMSG
3254+*
3255+*    THE FOLLOWING CODE IS PROVIDED TO GIVE SYSTEM
3256+*    IDENTIFICATION AT IPL TIME
3257+*
3258+IJBSUP20 CSECT
3259+          USING LTA,RF .        CONTINUE BASE USED IN TEMP CODE
3260+*
3261+          ORG   RES2311
3262+          BAL   R8,ZZIDCODE .   BRANCH TO PATCH CODE
3263+*
3264+          ORG   MVCEND .        EXTEND ONWARD INTO LTA
3265+ZZIDCODE DS    0D
3266+          BCTR  RE,0 .          EXECUTE OVERLAYED CODE
3267+          BCTR  RE,0 .          *
3268+*                               ADDED CODE STARTS HERE
3269+          MVC   CAW,ZZCCWADR .  CCW ADDRESS TO CAW
3270+ZZLOG    EQU   X'01F' .        1052 ADDRESS
3271+          SIO   ZZLOG .         TYPE THE MESSAGE
3272+          BNZ   ZZRETURN .      IF LOG DOWN, FORGET IT
3273+          TIO   ZZLOG
3274+          BC    2,*-4 .         LOOP WHILE TYPING
3275+ZZRETURN BR    R8
3276+*

3277+ZZCCW    CCW   09,ZZMSG,X'20',L'ZZMSG
3278+ZZCCWADR DC    A(ZZCCW)
3279+*
3280 ZZMSG    DC    C'DATACENTER  DOS-20  MOD 30  ASSGN SYSSLB'
3281          END
```

EXHIBIT 2

```
DATACENTER  DOS-20  MOD 30  ASSGN SYSSLB
0110A GIVE IPL CONTROL COMMANDS
set date= etc., etc.,
```

EXHIBIT 3

For IBM Internal Use Only

DOS/360 SUPERZOT

SUPERZOT, the DOS equivalent of the OS SUPERZAP, is one of those "I wish I had a" programs. It is used to replace absolute code in a phase in the CIL. The use of SUPERZOT may be preferable to the alternatives of linking or merging-in corrected code. (The code-to-be-corrected may have to be obtained by a CSERV/PUNCH, and REPing.) Existing code that is to be changed is verified before being replaced to help forestall errors. SUPERZOT is a CE Service Aid, available through FE Parts Centers as program 360Z-00-025. The instructions and write-up formerly in RETAIN are reprinted below for those who may have missed them. This item is in microfiche on PS EWS, card 24, ZS.24.0, frame A17.

SUPERZOT is mentioned in passing in INL 69-18, under 'Serviceability M Design Memo' on page 5, item V.A. (6).

It has been used to make PTF code changes to programs without having to relink the module changed. It has also been used to re-configure the system without re-assembling the supervisor when some 7 and 9 track tapes were interchanged.

INL articles concerning OS code modifications are usually given in terms of SUPERZAP changes. Now this facility is available to the DOS user as well.

DOS/360 SUPERZOT Service Aid 360Z00025.

ZUPERZOT is a program designed for use under DOS which closely resembles SUPERZAP-its big brother for OS. It allows conditional alterations of a core-image library phase(s) as it resides on disk, where SYSRES can be a 2311 or a 2314. Four control card types are accepted from SYSIPT:

1. NAME-defines phase name desired.
2. BASE (optional)-indicates offset address or relocation factor to be applied to a VERIFY or REP card address.
3. VERIFY- defines address of patch and contents.
4. REP-defines address of patch and replacement data.

More than one phase can be modified in each run. The job is completed when a /* card is detected. Several VERIFY and REP cards may be applied against any given NAME card, however a REP card must immediately follow its associated VERIFY card and must contain the same address. The length of data in both the VERIFY and REP cards must also be the same. The BASE card may be optionally included before the VERIFY and REP cards that it should be applied against. A new NAME card or a BASE card will reset any previously given BASE address. All control cards can be written in free format except the data portions of the VERIFY and REP cards. The data must be in 2-byte groups (4 hex characters) separated by a comma, but are not required to address any boundary alignment. A blank column acts as a delimiter. The address field of the BASE, VERIFY and REP cards can be 1 to 6 hex digits. All cards can utilize 80 columns. Continuation cards are not recognized. Comments may follow last field on any card after the blank delimiter.

Possible error messages printed on console are:

1. Invalid control card correct and EOB (unknown card type)
2. Phase not in library
3. Invalid verify card
4. Verify address outside phase
5. Verify data compare error
6. REP card out of sequence (not preceded by a verify)
7. Invalid REP card
8. Verify and REP lengths disagree
9. Verify and REP addresses disagree

Messages 2 through 9 will cause job to be cancelled.

SUPERZOT can be linkedited and executed in any partition with 6K using an ACTION F1 or F2 card. Program is not self-relocatable. Care should be exercised in its use in a multi-programming environment as no attempt is made to ensure other partitions are inactive or that a condense or reallocation could be in progress. Results under these conditions are indeterminate.

SAMPLE JOB STREAM:

The following job will modify the portion of SUPERZOT and restore it to original.

```
// JOB TEST SUPERZOT
// EXEC SUPERZOT
   NAME SUPERZOT
   BASE 3800 (relocation factor obtained
   from linkedit map)
VERIFY 810,5090,C676,9205
REP 810 1234,5678,9012
VERIFY.810 1234,5678,9012
REP 810 5090,C676,9205
/*
/&
```

S/360 MODEL 25 ICA NOTE ON DVCGENING

Although the Model 25 Integrated Communication Attachment reacts as if it were a 2703, it need not be considered exclusively as such.

ATS under DOS requires a 2702, and during initialization rejects any other device type code in the supervisor. For ATS, at least, the start-stop lines may be DVCGENed as 2702s.

This has been done at one Datacenter and ATS is running quite contentedly. While testing the possibility of using the Mod 25 for ATS, the lines were ADDed as 2702s with various MODE=(SADx) entries. There was no apparent effect on either DOS or ATS when running the gamut of possible SAD entries. The final system supervisor was assembled specifying 2702 lines, without including a MODE parameter, in the DVCGEN cards.

This works, of course, because the problem program, ATS, does not use any communications device-dependent channel commands. (Or if it does, the ICA has been designed not to react adversely.)

NOTE: A SECOM article 360A-CX-18X/076 advises that DOS ATS is being improved to run with 2701 and 2703 as control units also.

INFORMATION ABOUT THE NEWSLETTER

EVERY SALESMAN AND SYSTEMS ENGINEER SHOULD RECEIVE ONE COPY OF THE NEWSLETTER. Each FE Branch Manager should receive five copies for distribution to Customer Engineers.

The IBM Installation Newsletter is distributed only to IBM locations and is not available to customers. The Newsletter is separated into two parts. The first part contains information which is intended only for IBM personnel and the pages are labeled For IBM Internal Use Only. THESE PAGES, OR REPRODUCTIONS OF THEM, ARE NOT TO BE GIVEN TO CUSTOMERS. However, using your own good judgement, you may discuss pertinent information from these pages with a customer. The second part contains information which may be reproduced by the Branch Office at their discretion and given to customers. These pages are not labeled with any restrictive classification.

Note that when any abstracts for Type III and Type IV programs are given to the customer, he must be informed that the disclaimers for Type III and Type IV programs as contained in the Newsletter, applies. YOUR COOPERATION IN PROPERLY HANDLING THE CONTENT OF THE IBM INSTALLATION NEWSLETTER IS NECESSARY FOR ITS CONTINUED PUBLICATION.

Changes to the quantity of the Newsletter shipped are to be made by the Administration Manager. He should update and send the DP Administration Managers' Document Control card (Electro IBM N61508) to DP Document Control, DPD HQ.

Direct inquiries concerning the contents of this Newsletter to:

W. F. Malmborg, Editor
IBM Installation Newsletter
Dept. 812, DPD HQ
112 East Post Road
White Plains, New York 10601
Telephone 914 696-3781
Tie Line 254-3781

NEW PUBLICATIONS ANNOUNCED BY PRLs

Information is extracted and condensed from the weekly PRLs (Publications Release Letters) to insure that all Salesmen and Systems Engineers are aware of new or revised Reference Sources, Marketing Publications, and Field Engineering Publications. Normally, each issue of the Newsletter will contain information extracted and condensed from two PRLs, one following the other. The information will be placed in the Newsletter in its original sequence with no rearrangement of form numbers or titles. It is not intended to replace existing information and distribution sources. You should be certain that you are aware of these sources.

Do not use the condensed Newsletter information to order publications. Most but not all (for example, some films and slide sets) of the items are available from Mechanicsburg. To receive the items you need without delay, prior to ordering check the REQ (Requisition Form) code in the "Accumulative Supplement to IBM Publications Current Price List (Z20-0100)" which is attached to the weekly Publications Release Letter (PRL) and distributed to your Administration Manager. For an explanation of the REQ

codes and other publications information see the first few pages of the "IBM Publications Current Price List Z20-0100".

(IBM World Trade Corporation has its own distribution center and procedures.)

The IBM publications requisition (MO2-0618) or ITPS should be used for ordering items from the IBM Distribution Center, Mechanicsburg, Pa.

PRL #49 December 12, 1969

MARKETING PUBLICATIONS

FORM NO.	TITLE	DISPOSITION
G520-2223-0	TSS/360 for System/360 Model 6 Promotional Brochure	NEW
G570-0725-1	Wire Contact Relay Maintenance Manual	UES 570-0725-0
GC10-7005-0	IBM System/3 Disk System RPG II Reference Manual (Spanish Translation)	NEW
GC28-6395-1	IBM S/360 COBOL Differences: USA Standard COBOL Conversion	Scrap C28-6395-0
GC28-6534-2	IBM S/360 Operating System, Introduction	Scrap C28-6534-1
GC28-8201-2	IBM S/360 Operating System, PL/I Language Reference Manual	Scrap C28-8201-1, N33-6009, N33-6008, N33-6011
SH20-0698-0	General Purpose Simulation System/360 DOS Version 2 Operator's Manual Program Number 5736-XS1	NEW
SH20-0711-0	System/360 Text Processor- EDIT/360 Operations Manual	NEW
GK20-0243-1	Corrugated Box Plant Data Processing at Menasha Corporation	Scrap K20-0243-0
GN20-1993-0	TNL to IBM 1130 Rigid Frame Selection Program (H20-0580-0)	NEW
GN20-5400-1	TNL to PSM-S/360 OS PSI Re: 220-2004	NEW
GN27-3047-0	TNL to IBM S/360 Component Description: 2260 Display Station, 2848 Display Control Re: A27-2700-4	NEW
GN33-1532-0	TNL to IBM S/360 Model 20 Functional Characteristics Re: A26-5847-4, -5, -6	NEW
GN33-8061-0	TNL to IBM 1800 Time Sharing Executive System Operating Procedures Re: C26-3754-3	NEW
SR20-1094-0	1410/7010 Data Processing System Programming Workbook	NEW
SR20-1095-0	1410/7010 DPS Programming Video Course Examinations	NEW
SR20-1098-0	7090/7094 DPS Programming Video Education Examinations	NEW
SR20-1099-0	7090/7094 DPS Programming Advisor Guide	NEW
SR20-1106-0	S/360 DOS/TOS CALL/SAVE/Return Techniques Student Materials	NEW
ZR20-6010-0	System/3 Introduction to Computing Systems Education Guide	NEW
GR20-9321-0	1410/7010 DPS Programming Video Tape Course Description	NEW
SY20-0161-4	Service for Consultants Manual	Scrap Y20-0161-3, Y20-0458 Y20-0472 & Y20-0485
SY20-0491-0	TNL to Service for Consultants Manual Re: Y20-0161-4	NEW
LY20-0483-0	General Purpose Simulation System/360 DOS Version 2 System Manual Program Number 5736-XS1	NEW
GY21-0506-0	IBM System/3 Card Systems Utilities PLM	NEW
ZZ77-9216-0	Emulating a 1311 under the 1410 Emulator/50 of COS/50	NEW

REFERENCE SOURCES

G320-1031-0	BOIS - Branch Office Information Sources Brochure	NEW
Z320-1032-0	BOIS Master Abstract Catalog	NEW
Z320-1033-0	BOIS Display Stand	NEW
Z320-1034-0	BOIS Master Index Tabs	NEW
Z320-1046-0	KWOC - Keyword Out of Context	NEW
Z320-1048-0	KWOC - Keyword Out of Context - WTC Section	NEW
ZZ20-0100-24	IBM Publications and Programs Current Price List	Scrap Z20-0100-23

PRL #50 December 12, 1969

REFERENCE SOURCES

FORM NO.	TITLE	DISPOSITION
GN20-0361-40	S/360 Model 20 SRL Newsletter Re: A26-3565-5	Scrap N20-0361-39

MARKETING PUBLICATIONS

G221-0352-1	IBM 1062 Tellar Terminal Proposal	UES 221-0352-0
G221-0432-2	IBM 1287 Proposal Insert	Scrap 221-0432-1
G221-0482-1	LPS/1130 Proposal Insert	Scrap 221-0482-0
G221-0522-0	IBM 1255 Magnetic Character Reader Proposal Insert	NEW
G320-1035-0	Improving School of Business Courses Through the Use of the Computer	NEW
G321-0019-0	IBM Systems Journal Vol. 8 No. 4	NEW
G505-0022-0	HEXAPAWN Promotional Folder	NEW
G520-2006-1	Graphic Data Processing - A Dramatic Development in Handling Information	Scrap 520-2006-0
G520-2191-1	How to Get More Time to Spend	Scrap 520-2191-0
G520-2207-0	ATS/360 Promotional Brochure	NEW
G520-2241-0	ITF - Interactive Terminal Facility	NEW
G520-2242-0	ITF - The Interactive Terminal Facility Time Sharing Power for System/360	NEW
G520-2256-0	Time Sharing TSO for OS/360 - Promotional Brochure	NEW

PRL #50 (Continued)

G520-2275-0	TSO - Time Sharing Option for OS/360 - Promotional Flyer	NEW
G520-2276-0	Time Sharing Marketing Pocket Kit	NEW
S570-0256-0	Care and Handling of IBM Magnetic Tapes	NEW
GA22-6932-0	IBM 545 Models 3 and 4 Output Punch Reference Manual	NEW
GA26-5847-6	System/360 Model 20 Functional Characteristics	UES A26-5847-5
GA26-5916-5	IBM 1130 SRL Bibliography	Scrap A26-5916-4
GA27-2706-1	IBM 7770 Audio Response Unit Model 3: OEMI	Scrap A27-2706-0 & N27-3008
GA27-3027-0	IBM 7772 Audio Response Unit Other Equipment Manufacturer's Information	NEW
GB20-0070-1	Piecework Payroll for the Apparel Industry Application Manual	UES B20-0070-0
GC21-5001-1	IBM S/360 BPS Distribution Program	Scrap C21-5001-0 & N33-8586
GA19-0020-1	IBM 3976 Modem Models 1 & 2 Component Description	Scrap A19-0020-0
GC21-7526-0	Card System Sort/Collate Reference Manual	NEW
GC21-7538-1	IBM System/3 Data Recording/Data Verifying Program Operator's Guide	Scrap C21-7538-0
GC26-3734-0	IBM System/360 Operating System Time Sharing Option TSO Assembler	NEW
GC28-6115-1	IBM 7070 Series Generalized Merging Program Merge 91	NEW
GC28-6404-1	IBM S/360 OS: Time Sharing Option COBOL Prompter	Scrap C28-2404-0
GC28-6698-0	IBM S/360 OS: Time Sharing Option, Planning for TSO	NEW
GC28-6824-0	IBM S/360 OS FORTRAN Program Products for OS and OS with TSO	NEW
GC28-6825-0	IBM S/360 OS Disk Operating System Interactive Terminal Facility:PL/I	NEW
GC28-6826-0	IBM S/360 OS System/360 Disk Operating System Interactive Terminal Facility: Basic	NEW
GC28-6827-0	IBM S/360 OS Time Sharing Option Interactive Terminal Facility: PL/I	NEW
GC28-6828-0	IBM S/360 OS Time Sharing Option Interactive Terminal Facility: Basic	NEW
GC28-6844-0	IBM S/360 OS FORTRAN IV Library Mod I	NEW
GE20-0188-4	Retail Impact - Application Description Manual	UES E20-0188-3
GH20-0186-3	General Purpose Simulation System/360 Application Description Manual	UES H20-0186-2
GH20-0318-3	Customer Information Control System Application Description Manual	UES H20-0318-2
GH20-0498-1	Property and Liability Information System Basic Operations Manual	Scrap H20-0498-0
GH20-0554-1	S/360 Model 20 Bill of Material Processor Operations Manual	UES H20-0554-0
GK20-0334-0	The IBM 1800 Controls a Metal Cutting Transfer Line	NEW
GK20-0349-0	IBM S/360 Mod 20 Teleprocessing System at Safeco Corporation	NEW
GN20-2016-0	TNL to S/360 Requirements Planning Application Description Manual Re: H20-0487-2	NEW
GN33-8582-0	TNL to BPS Autotest Specifications Re: C24-3343-2	NEW
ZR20-1038-1	S/360 DOS/TOS Data Management Coding Class Exercises	Scrap R20-1038-0
ZR20-6002-0	System/3 Installation Control for Management	NEW
GR20-9381-0	System/360 Generalized Information System Course Description	NEW
SR29-0019-2	Fundamentals of Programming PI Text	UES R29-0019-1
SR29-0103-3	FORTRAN for the 1130 Chapter 3 PI	UES R29-0103-2
GV20-0234-0	TSO Time Sharing Option for OS/360 Slides	NEW
GV20-0235-0	The Life and Times of Tom Shearing (A Time Sharing Overview)	NEW
ZV2S-8218-0	S/360 Model 20 Installation Programming - Card - Learner - Paced Format Tape 1 : Introduction, Punched Card Concepts - Video Tape	NEW
ZV2S-8219-0	S/360 Model 20 Installation Programming - Card - Learner - Paced Format Tape 2 - Computer Concepts, Introduction to RPG - Video Tape	NEW
ZV2S-8220-0	S/360 Model 20 Installation Programming - Card - Learner - Paced Format Tape 3 - RPG Internal Data Handling, RPG Table Look-Up - Video Tape	NEW
ZV2S-8221-0	S/360 Model 20 Installation Programming - Card - Learner - Paced Format Tape 4 - RPG Move Operations - Part I; RPG Move Operations - Part II	NEW
ZV2S-8222-0	S/360 Model 20 Installation Programming - Card - Learner - Paced Format Tape 5 - RPG Logic, RPG Matching Records - Video Tape	NEW
ZV2S-8223-0	S/360 Model 20 Installation Programming - Card - Learner - Paced Format Tape 6 - Punched-Card Utility Programs, Installation Control, Planning An Application - Video Tape	NEW
ZV2S-8224-0	S/360 Model 20 Installation Programming - Card - Learner - Paced Format Tape 7 - Flowcharting and Programming, Planning a Program	NEW
ZV2S-8225-0	S/360 Model 20 Installation Programming - Card - Learner - Paced Format Tape 8 - Overview - Video Tape	NEW
ZV2S-8227-0	System/3 Installation Control for Management - A review of System Design Procedures - Video Tape	NEW
ZY23-8218-0	IBM System/360 Hardware Overview - Video Tape	NEW
ZY23-8223-0	ISAM Internals - Reel 1 - Overview - Video Tape	NEW
ZY23-8224-0	ISAM Internals - Reel 2 - Load Mode - Video Tape	NEW
ZY23-8225-0	ISAM Internals - QISAM (Scan Mode) Video Tape	NEW
ZY23-8226-0	ISAM Internals - Reel 4 - Data Set Integrity ISAM & Timing	NEW
ZY23-8228-0	Price of Admission - Video Tape	NEW
ZY23-8230-0	Terminals for Industry Marketing (IBM Version) Video Tape	NEW
ZY23-8231-0	FORTRAN Review - Video Tape	NEW
ZY23-8232-0	User Terminal 1050 and Model 33 Telewriter - Video Tape	NEW
ZY23-8233-0	Management Science - Reel 1 - Introduction to Management Science Resource Allocation - Video Tape	NEW
ZY23-8234-0	Management Science - Reel 2 - VSP Part I - Video Tape	NEW
ZY23-8235-0	Management Science - Reel 3 - Forecasting Part I and II - Video Tape	NEW
ZY23-8236-0	Management Science - Reel 4 - Simulation Part I and II - Video Tape	NEW
ZY23-8237-0	Management Science - Reel 5 - Project Management Implementation Video Tape	NEW
ZY23-8238-0	IBM S/360 DOS Demonstration Video Tape	NEW
ZY23-8282-0	APL/360 An Interactive Approach - Reel 1 - Capabilities of APL	NEW
ZY23-8283-0	APL-360 An Interactive Approach - Reel 2 - Elementary Arithmetic Operations - Video Tape	NEW
ZY23-8284-0	APL/360 An Interactive Approach - Reel 3 The Concept of Reduction More About the Order of Execution in APL - Video Tape	NEW

PRL #50 (Continued)

ZY23-8285-0	APL/360 an Interactive Approach - Reel 4 - Standard Scalar Monadic f Functions, a Prepared Exercise in APL Functions - Video Tape	NEW
ZY23-8286-0	APL/360 an Interactive Approach - Reel 5 - Function Definition - The Syntax of Functions - Video Tape	NEW
ZY23-8287-0	APL/360 an Interactive Approach - Reel 6 - Function Editing - Global and Local Variables - Video Tape	NEW
ZY23-8288-0	APL/360 an Interactive Approach - Reel 7 - Personal Library Management, Video Tape	NEW
ZY23-8289-0	APL/360 an Interactive Approach - Reel 8 - The Mixed Monadic Functions Index Generator and Dimension Vector - The Mixed Functions Catenate, Ravel, Restructure - Video Tape	NEW
ZY23-8290-0	APL/360 an Interactive Approach - Reel 9 - Literal or Character Values More Mixed Functions, Ranking Indexing, Video Tape	NEW
ZY23-8291-0	APL/360 an Interactive Approach - Reel 10 - Video Tape	NEW
ZY23-8292-0	APL/360 an Interactive Approach - Reel 11 - Base Value and Representation - Video Tape	NEW
ZY23-8293-0	APL/360 an Interactive Approach - Reel 12 - Branching, Trand and Stop Controls; Recursion - Video Tape	NEW
ZY23-8294-0	APL/360 an Interactive Approach - Reel 13 - I - Beam Functions - Multidimensional Arrays - Video Tape	NEW
ZY23-8395-0	APL/360 an Interactive Approach - Reel 14 - Operations on Multidimensional Arrays - Transposition - Video Tape	NEW
ZY23-8396-0	APL/360 an Interactive Approach - Reel 15 - The Generalized Outer Product - The Generalized Inner Product - Video Tape	NEW
ZY23-8397-0	APL/360 an Interactive Approach - Reel 16 - Applications of APL to Matrix Algebra - Input and Output - Video Tape	NEW
ZY23-8398-0	APL/360 an Interactive Approach - Reel 17 - Miscellaneous System Commands Review and Overview - Video Tape	NEW
ZY23-9642-0	Basic PL/I Workshop - Reel 1 - Data Types and I/O Review - Video Tape	NEW
ZY23-9643-0	Basic PL/I Workshop - Reel 2 - Compiler Output - Video Tape	NEW
ZY23-9644-0	Basic PL/I Workshop - Reel 3 - Devined Attributes - Pointers and Based Variables - Video Tape	NEW
ZY23-9645-0	Basic PL/I Workshop - Reel 4 - Indexed-Sequential Files - Video Tape	NEW
ZY23-9646-0	Basic PL/I Workshop - Reel 5 - Sub-Routines & Functions - Video Tape	NEW
ZY23-9647-0	Basic PL/I Workshop - Reel 6 - Scope of Identifiers - Video Tape	NEW
GY26-5797-2	IBM 1500 Configurator	Scrap Y26-5797-1
Z20-1955-0	Pulp and Paper Marketing Guide	NEW
ZZ77-9218-0	Converting 1440 Disk Files to 360/25 Disk Files for Use Under Hardware	NEW
ZZ77-9219-0	Programming Hints and Installation Aids for the 1259 MICR Reader Sorter	NEW
ZZ77-9227-0	S/360 OS BTAM BSC error Recovery Procedures	NEW
ZZ77-9228-0	A Systems Approach to Fully Automated Country Club Billing Without Manual Keypunching	NEW

FIELD ENGINEERING

S123-0486-2	IBM 1810 IPC	Scrap 123-0486-2 & S27-0657
S123-1001-4	IBM 2948 Display Control Unit Model 4 IPC	Scrap 123-1001-3
S131-0010-1	Keyboards Elastic Diaphragn Switch Encoded IPC	NEW
S223-6727-0	IBM 824 826 Typewriter Card Punch FETOM	NEW
S227-5712-1	IBM 1621/1624 FEMM	UES 227-5712-0 & S27-5830
S229-0022-3	FE Service Code Folder	UES 229-0022-2
G229-0032-0	College Recruiting Brochure	NEW
SR23-3086-2	CAI Model 40 Slide Illustrated Student Guide	Scrap R23-3086-1
SR25-5440-0	SLT EC Book of Illustration Supplementary Course Material	NEW
SR27-9608-1	2020 PU Submod 5 Lab for FIS	Scrap R27-9608-0
SS23-9056-0	2044 Processing Unit Parts Catalog Supplement Re: 123-0455-2	NEW
SV31-0184-1	Monolithic System Technology Microfiche Card	NEW
SY22-6814-0	Supplement to 2091 Processing Unit FEMM Re: Y22-6659-1, -2	NEW
SY24-0521-0	Supplement to 2560 Model 20 FEMDM Re: Y24-3529-1	NEW

December 19, 1969 **IBM INSTALLATION NEWSLETTER** Issue No. 69-25

Table of Contents

OS/360 MULTIPLE CONSOLE SUPPORT 2740 CHARACTER SET

The 2740 support under OS/360 MCS requires the EBCDIC character set, feature #9571. All installations planning to use this support should be advised to insure that they order the correct feature.

OS/360 ADDING START COMMAND PROCEDURES

This contribution has not been submitted to any formal IBM test. Potential users should evaluate its usefulness in their own environment prior to implementation.

In OS, START commands reference cataloged procedure names such as RDR, WTR, etc. Programs which are executed as a result of a start command must be listed in a module of linklib. This module name is IEEVLNKT. The contents of IEEVLNKT is illustrated in figure one with the name IEBGENER added to the bottom of the list. IEBGENER was added so that two new cataloged procedures, CTC and CTP, could be added to the system (see coding examples in figures two and three). These cataloged procedures allow the operator to do card-to-card and card-to-printer operations easily. He merely specifies:

```
    s ctc.s        for card-to-card
or  s ctp.s        for card-to-printer.
```

This technique might be used for OLTEP (reference INL No. 69-21,) by adding IFDOLT to IEEVLNKT and adding a simple cataloged procedure something like:

```
//JOBLIB DD DSN=sys1.OLTLIB,disp=old
//step1 exec pgm=IFDOLT
//diagmsg DD sysout=A
```

with the name of OLTEP.

CTC and CTP have been used on an OS Release 17 MFT system. It has not been tried with OLTEP.

```
//IEEVNLKT JOB 251,PETERSON,MSGLEVEL=1
//STEP1 EXEC ASMFCL
//ASM.SYSIN DD     *
IEEVLNKT   CSECT
REFTBL     EQU *
           DC  CL8'IEFIRC'
           DC  CL8'IEFSD080'
           DC  CL8'IEEVMNT2'
           DC  CL8'IHKRJBGN'
           DC  CL8'IEFVRRC'
           DC  CL8'IKDINPRO'
           DC  CL8'IKDSGJP'
           DC  CL8'IEFVMA'
           DC  CL8'IKAGFX'
           DC  CL8'IKAGJP'
           DC  CL8'IEFSD060'
           DC  CL8'IEBGENER'
           DC  X'00'
           END
/*
//LKED.SYSLMOD DD DSNAME=SYS1.LINKLIB
(IEEVLNKT),DISP=OLD
/*
```

Figure 1

```
./     ADD LIST=ALL,LEVEL=01,SOURCE=0,
           NAME=CTC
//STEP1        EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSIN DD DUMMY
//SYSUT1 DD UNIT=00A,DCB=(RECFM=F,
           BLKSIZE=80,LRECL=80)
//SYSUT2 DD UNIT=00D,DCB=(RECFM=F,
           BLKSIZE=80,LRECL=80)
```

Figure 2

```
./     ADD LIST=ALL,LFVEL=01,SOURCE=0,NAME=
           CTP
//STEP1    EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSIN DD DUMMY
//SYSUT1 DD UNIT=00A,DCB=(RECFM=F,BLKSIZE=
           80,LRECL=80)
//SYSUT2 DD SYSOUT=A,DCB=(RECFM=FB,LRECL=
           80,BLKSIZE=1600)
```

Figure 3

OS/360 COBOL F CLIST ADDITIONS

This contribution has not been submitted to any formal IBM test. Potential users should evaluate its usefulness in their own environment prior to implementation.

The following technique may be of interest to OS Release 17 COBOL F users.

The CLIST option does not produce global tables, literal pools, or register assignments. Three TM instructions were changed and 2 patches made to the COBOL compiler to produce these items along with the condensed listing.

The DELINK 0 program was run to provide space for the patches and then re-linked. Following this, SUPERZAP was run to make the changes and additions.

These modifications have been tested only for Release 17. Subsequent releases must be checked for applicability.

The JCL used is shown in Exhibit 1 following:

```
//GO EXEC PGM=DELINK0
//SYSPRINT DD SYOUT=A
//SYSPUNCH DD DSN=&TEMP(A),DISP=(NEW,
 PASS),UNIT=SYSDA,
//                    SPACE=(80,(100,50,1))
//SYSLIB DD DSN=SYS1.LINKLIB,DISP=SHR
//SYSIN DD *
> IEQCBL60 IEQ604 X'F1C'

//LKED EXEC PGM=IEWL,PARM='XREF,LET,LIST,
 NCAL'
//SYSLMOD DD DSN=SYS1.LINKLIB,DISP=OLD
//SYSPRINT DD SYSOUT=A
//SYSPIN DD DSN=*.GO.SYSPUNCH,DISP=(OLD,
 DELETE)
//SYSUT1 DD UNIT=2314,SPACE=(TRK,(50,50))
//SYSLIN DD *
 INCLUDE SYSPIN(A)
 INCLUDE SYSLMOD (IEQCBL60)
 ENTRY IEQ601
 NAME IEQCBL60(R)
//ZAP EXEC PGM=SUPERZAP,REGION=60K
//SYSPRINT DD SYSOUT=A
//SYSLIB DD DSN=SYS1.LINKLIB,DISP=OLD
//SYSIN DD *,DCB=BLKSIZE=80
NAME IEQCBL60 IEQ602
BASE 0C50
VERIFY 14CC 9110
REP 14CC 9112
NAME IEQCBL60 IEQ603
BASE 1558
VERIFY 15D6 4460,8098
REP       15D6 47F0,AEF2
VERIFY 2546 58F0,C3E4
REP       2546,47F0,AEE0
NAME IEQCBL60 IEQ604
BASE 2558
VERIFY 3438 40404040404040404040404040404040
               4040404040404040

 REP 3438 9110915B
 REP 343C 47808FF4
 REP 3440 58F0C3E4
 REP 3444 05EF
 REP 3446 47F08FF4
 REP   344A 9110915B4780808C446080984
                7F0808247F0808C
 NAME IEQCBL60 IEQ606
BASE 4518
VERIFY 4F30 9110
REP       4F30 9112
NAME IEQCBL60 IEQ607
BASE 51B0
VERIFY 548C 9110
REP       548C 9112
```

DOS/360 ONE CARD READER SERVING TWO PARTITIONS

This contribution has not been submitted to any formal IBM test. Potential users should evaluate its usefulness in their own environment prior to implementation.

The ASSGN1 DOS supervisor generation macro, described in INL 68-25, can be used to permanently assign a card reader to more than one partition as SYSIPT, SYSRDR, etc. Jobs can be started through the reader, and when executing, another job can be begun in another partition through the same reader. To prevent Job Control, upon EOJ, from flushing out the reader looking for a /&, the PAUSE command (no slashes, and with the partition I.D.) is employed. Using this approach, Job Control comes to the console with the 'READY FOR COMMUNICATIONS' message at program EOJ. Typical job stream in the reader:

```
//JOB BACKGRND      JOB A
 PAUSE BG
//EXEC 0601
//JOB FOREGRND      JOB B
 PAUSE F2          (or F1)
//EXEC 0700
```

In addition to SYSPIT and SYSRDR for both partitions, it is recommended that SYS004 also be assigned to the reader for IBM utitlities and COBOL. With this installed, no // ASSGN statements for the reader will be accepted by JOB CONTROL - all will result in 'CONFLICTING I/O ASSIGNMENT' messages, therefore users should read input data cards on either SYSIPT (ALC) or SYS004 (COBOL) and not employ // ASSGN cards for the reader.

Example of console operation to get the sample job stream going:

```
BG JOB BACKGRND         JOB A
  PAUSE BG
AR READY FOR COMMUNICATIONS
AR BATCH F2
F2
F2 JOB FOREGRND         JOB B
  PUASE F2
BG READY FOR COMMUNICATIONS     (EOJ)
BG /& (ENTERED BY OPERATOR)
BG EOJ BACKGRND
BG STOP
F2 READY FOR COMMUNICATIONS (EOJ)
F2 /& (ENTERED BY OPERATOR)
F2 EOJ FOREGRND
F2 STOP
```

If one of the partitions sharing the reader is using it to read input data cards and the operator wants to start another program in the other partition, he may do so by entering the JOB CONTROL statements through the console, but if a PAUSE is not included, any /&, or // JOB not preceded by a /& since the last // JOB, will post an EOJ in the reader pub and cancel the program that is reading input data cards with a 'READING PAST /&' message.

This technique has been used with Release 20 of DOS.

DTR PUNCHING WITH DOS UTILITY

This contribution has not been submitted to any formal IBM test. Potential users should evaluate its usefulness in their own environment prior to implementation.

The manual of operations for the Card Deck Retriever Program supplied with a shared Distribution Tape Reel (DTR) with Type III or IV programs ordered from PID indicates that other methods can be used to punch the card decks.

The following is a method using a DOS utility:

```
// JOB TAPECARD DOS/360 TAPE TO CARD
   UTILITY
// ASSGN SYS004, X'180', X'C0'
// MTC REW, SYS004
// MTC FSF, SYS004, NN
   WHERE NN=2+3 TIMES NO. CARD DECKS
   SKIPPED
// ASSGN SYS006, X'00A'
// UPSI 10000000 NO INPUT LABELS
// EXEC TPCD
// UTC TR,FF,A=(80,1600), B=(80,80),
   IN, R1, S1, O1
// END
/* END OF DATA
/& END OF JOB
```

The operator response to the console message:

8001D IS IT EOF

is: Y EOB

Use the following // MTC cards:

to retrieve the first card deck (N=0)

// MTC FSF,SYS004,2

To retrieve the fourth card deck (N=3)

// MTC FSF,SYS004,11

1130 RPG STACKER SELECT DURING CALCULATIONS

This contribution has not been submitted to any formal IBM test. Potential users should evaluate its usefulness in their own environment prior to implementation.

Input card records can only be selected to the alternate stacker on "record type" criteria. If it is desired to stacker select an input card record on say, an indicator being set on during calculation, the only recourse is to describe the file as a combined file and "false" punch the card to use the stacker select option on output. This is undesirable because punching speed is significantly slower than reading.

The following coding example represents an assembly language subroutine (and associated control cards) which will permit input cards to be selected to the alternate stacker during calculations.

```
//ASM
            ENT     STACK
  IOCC      DC      0
            DC      /1480
  STACK     DC      * - *
            XIO     IOCC
            BSC I   STACK
            END

// DUP
 * STORE   WS   UA STACK
```

Example:

An input record is chained to a disk file. The accessed disk record contains field PAT which is tested for being less than 200.00. If it is, indicator 04 is turned on. To select the card which accessed this record to the alternate stacker, the following line is written in the calculations section:

```
C   04      EXIT    STACK
```

This technique has been tested and used by the author.

TYPE III AND IV PROGRAMS

Type III programs are those which have been submitted by one or more IBM employees. They are programs of general interest submitted for unrestricted distribution. They have met a basic set of programming and documentation standards but are not program tested in any formal fashion by the IBM Corporation. The user is expected to make the final evaluation as to their usefulness in his own environment. These programs were available for delivery from the Program Information Department prior to June 23, 1969.

Type IV programs are those contributed for unrestricted distribution by one or more authors of which at least one is an employee of an IBM customer. They are made available by IBM essentially in the author's original form, but conform to published Type IV standards. IBM exercises no control over the technical content of the documentation but merely assures that the quality of reproduction is satisfactory. Type IV programs have not been tested by IBM. The user is expected to make the final evaluation as to their usefulness in his own environment.

IBM MAKES NO WARRANTY, EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE AS TO THE DOCUMENTATION, FUNCTION OR PERFORMANCE OF THESE PROGRAMS.

TYPE III PROGRAMS WITH SERVICE A CLASSIFICATION

Type III programs which were given Service A Classification, perform functions which may be fundamental to the operation and maintenance of the user's system. These programs have not been subjected to formal test by IBM.

Until reclassified, IBM will provide for these Type III programs with the following: (a) Central Programming Service including design error correction and automatic distribution of corrections; (b) Field Engineering Programming Service including design error verification, Authorized Programming Analysis Report (APAR) documentation and submission, and application of Program Temporary Fixes or development of an emergency by-pass when required.

IBM does not guarantee service results or represent or warrant that all errors will be corrected. The user is expected to make the final evaluation as to the usefulness of these programs in his own environment.

THE FOREGOING IS IN LIEU OF ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

---------- NEW TYPE IV PROGRAM ----------

OS MASTER SCHEDULER INITIALIZATION IMPROVEMENTS

DESCRIPTION - These modifications to the MFT and MVT Master Scheduler Initialization modules cause improved generation of automatic commands. In addition, messages are generated prompting the operator, in situations where he would otherwise be waiting for a wait state. The modifications are for OS/360 Release 17 or later.

PROGRAMMING SYSTEMS - Written in S/360.0S ALF; operates under control of OS/360 MFT or MVT.

MINIMUM SYSTEM REQUIREMENTS - These modifications require a level G (128K) or larger CPU for MFT, and a level H (256K) or larger CPU for MVT. They have been tested for both MFT and MVT on a 360/40H.

BASIC PROGRAM PACKAGE
DOCUMENTATION - Write-up.
MACHINE READABLE - Source code and sample problem.

OPTIONAL PROGRAM PACKAGE - None.

ORDERING INFORMATION: PROGRAM NUMBER 360D050003

	PROGRAM NUMBER EXTENSION	DISTRIBUTION MEDIUM TYPE	CODE	USER VOLUME REQUIREMENT
BASIC	none	MT 9/1600	29	01
		MT 9/800	28	01
		MT 7DC/800	26	01
OPTIONAL	none	none		none

---------- NEW TYPE IV PROGRAM ----------

CDUTL, A GENERAL PURPOSE CARD UTILITY

DESCRIPTION - CDUTL is a general purpose card utility which will reproduce, gang punch, enter, sequence and ID, and allow the key board to be used as a card punch. Program will reproduce binary decks.

PROGRAMMING SYSTEM - Written in BAL it will operate with either V1 or V2 of the 1130 Disk Monitor.

MINIMUM SYSTEM REQUIREMENTS - Same as those required for the 1130 Disk Monitor.

BASIC PROGRAM PACKAGE
DOCUMENTATION - Write-up.
MACHINE READABLE - Object code.

OPTIONAL PROGRAM PACKAGE - None.

ORDERING INFORMATION: PROGRAM NUMBER 1130000011

	PROGRAM NUMBER EXTENSION	DISTRIBUTION MEDIUM TYPE	CODE	USER VOLUME REQUIREMENT
BASIC	none	CARDS	15	none
OPTIONAL	none	none		none

---------- NEW TYPE IV PROGRAM ----------

FAST

"FAST" A FORTRAN Program for the Defitherm Process. The major cause of unlevelness in the dyeing of acrylic fibers is the presence of temperature differences in the dyebath during the heating-up phase. In the Defitherm Process the exhaustion rate of Basacryl dyes is controlled thermally to ensure production of level dyeings. To accomplish this, dyeings must be controlled at a certain temperature which is dependent on the following: DYESTUFF, SHADE DEPTH, FIBER, DYEING EQUIPMENT. It is the purpose of "FAST" to determine this dyeing temperature in addition to the amount of Thermoregulator (Defithermol TR) required. Because of the best exhaustion rate for various dyeing equipment must be determined empirically, the program does compute this variable but allows the dyer the option of selecting a dyeing time of 15, 30, 45, 60, 90, or 120 minutes. In this manner the dyer can optimize on the time parameter basing his decisions on cost and the available dyeing equipment.

The single input card to the program includes dyestuff numbers and corresponding concentrations for a given formula. The desired dyeing time may also be selected on this card.

PROGRAMMING SYSTEM - Written in FORTRAN and operates using the 1130 Disk Monitor.

MINIMUM SYSTEM REQUIREMENTS - An 8K 1130, 1442 Card Read Punch, 1132 Printer. Common 248, Variables 198, Program 1236. Disk Requirements; Sectors 15. System may be operated without benefit of the 1132 Printer by changing the variable NOUT in the Mainline program to A 1 rather than 3.

BASIC PROGRAM PACKAGE
DOCUMENTATION - Write-up.
MACHINE READABLE - Source code and sample problem.

OPTIONAL PROGRAM PACKAGE - None.

ORDERING INFORMATION: PROGRAM NUMBER 1130173004

	PROGRAM NUMBER EXTENSION	DISTRIBUTION MEDIUM TYPE	CODE	USER VOLUME REQUIREMENT
BASIC	none	CARDS	15	none
OPTIONAL	none	none		none

ACCUMULATIVE LIST OF NEW TYPE III AND IV PROGRAMS

Following is an accumulative list of new Type III and Type IV programs which have been made available since the last edition of the Catalogs of Programs or their Supplements. Abstracts describing these programs can be found in the Installation Newsletter Issue No. shown below.

Program Number	Program Title	Newsletter Issue No.
360D-00.0.017	SPOOL, A PACKAGE TO CONVERT CURRENT OR FUTURE PRINTED REPORTS TO MAGNETIC TAPE FOR FOREGROUND OR BACKGROUND PRINTING LATER	69-09
360D-00.0.020	2495 UTILITY PROGRAM FOR OS/360	69-11
360D-00.0.022	PUFER - PUNCH FEED READ UTILITY CARD TO TAPE	69-15
360D-00.1.021	S/360 GENERALIZED EXTRACTION AND MATCH PROGRAM	69-07
360D-00.1.022	NYU PRINT-UTILITY PROGRAM	69-07
360D-02.0.004	INDEX SEQUENTIAL ERROR BYTE CHECK ROUTINE	69-24
360D-00.2.006	OS SYSTEM/360 FLOWCHART (OS FLOW): DOS FLOWCHART UNDER OS/360	69-03
360D-00.3.025	S/360 DOS TAPE TO PRINTER/PUNCH SPOOLING SYSTEM	69-08
360D-00.3.026	FORTRAN TAPE DISK READ WRITE SUBROUTINES	69-16
360D-00.3.027	SEVEN TRACK OCTAL TAPE DUMP	69-11

360D-00.3.028	1400 INPUT/OUTPUT STANDARD LABEL PROCESSING EXITS FOR 483 DISK & TAPE SORT	69-19
360D-00.3.029	TAPE FILE EXTENSION MACRO - INSTRUCTION	69-20
360D-00.4.015	DAP DIRECT ACCESS PATCH PROGRAM	69-09
360D-00.4.018	IBM SYSTEM/360 INDEXED SEQUENTIAL DASD TO PRINT UTILITY	69-17
360D-00.5.005	FORMATTED LISTING OF VTOC FOR OS/360	69-08
360D-00.5.006	SHAREDISK	69-11
360D-00.6.011	A HYPERTEXT EDITING SYSTEM FOR THE S/360 USING THE 2250 DISPLAY	69-18
360D-01.0.009	OS/360 DATE AND TIME SUBROUTINE (DT01) WITH CONVERSION FROM JULIAN DATE TO GREGORIAN	69-09
360D-01.1.006	COMPILE AND GO FOR THE OS 360, PL/I (F) COMPILER	69-06
360D-01.3.002	1287 CHECK DIGIT AND NUMERIC RECONSTRUCTION MACROS	69-11
360D-01.4.010	SYSGEN AIDE: DIRECTORY DUMP	69-16
360D-03.0.007	S/360 MOD 44 MULTIPROGRAMMING SYSTEM	69-08
360D-03.0.008	DOS IOCS LOGIC MODULE SOURCE STATEMENT GENERATOR	69-08
360D-03.0.010	STENO TO ENGLISH TRANSLATION	69-17
360D-03.0.012	PARS CONTROL (ACP)	69-13
360D-03.1.014	FAST ASSEMBLER - INTERPRETER FOR SYSTEM/360 ASSEMBLER LANGUAGE PROGRAMS (VERSION 1)	69-14
360D-03.2.011	OS/360 PL360 COMPILER	69-06
360D-03.2.013	BASIC COMPILER UNDER TSS/360	69-13
360D-03.2.014	THE SIMSCRIPT II PROGRAMMING LANGUAGE	69-16
360D-03.2.015	THE XPL COMPILER GENERATOR SYSTEM	69-17
360D-03.3.008	COURSE III/OS A REAL-TIME COMPUTER ASSISTED INSTRUCTIONAL SYSTEM	69-01
360D-03.3.009	1404 IOCS MACROS FOR DOS/TOS	69-09
360D-03.3.010	SNAP PROCESS OR (PROTOTYPE)	69-10
360D-03.3.011	COMIT/360	69-15
360D-03.4.026	BSAM SUBROUTINE PACKAGE FOR OS 360, VERSION I	69-06
360D-03.4.027	FORTRAN RANDOM I/O SUBROUTINE	69-03
360D-03.4.028	2260 LOCAL TERMINAL USERS SYSTEM (LOTUS)	69-06
360D-03.4.031	KNUTH PROPOSAL INPUT/OUTPUT PACKAGE (KIOP)	69-16
360D-03.6.009	OS/360 COBOL CROSS-REFERENCE AND RESEQUENCING PROGRAM	69-05
360D-03.6.010	JOB STREAM MANAGER OS/MVT	69-03
360D-03.6.012	PROGRAM FOR THE CREATION OF CROSS REFERENCE TABLES, FROM S/360 RPG SOURCE PROGRAMS, FOR INDICATORS AND FIELD NAMES	69-10
360D-03.6.013	SYSTEM MANAGEMENT ROUTINES OS/MVT	69-10
360D-03.6.017	PL/60	69-15
360D-03.6.018	NEATER: A PL/I SOURCE STATEMENT REFORMATTER	69-17
360D-03.7.025	CARRIER RETURN/LINE FEED ON THE 1052/53, MISLCRLF	69-03
360D-03.7.026	DCLC A DOS/TOS/OS ASSEMBLER MACRO TO DEFINE A LOWER-CASE CHARACTER CONSTANT - DEFINE-CONSTANT-LOWER-CASE	69-09
360D-03.7.027	EDIT MACRO FOR DOS, TOS, OR OS ASSEMBLY LANGUAGE, S/360	69-07
360D-03.7.028	DOS MACRO FOR TYPEWRITER INPUT/OUTPUT	69-10
360D-03.7.031	DOS MACROS TO DUMP ISAM FILES FROM 2311 OR 2314 TO TAPE AND RESTORE FROM TAPE TO 2311 OR 2314	69-15
360D-03.8.013	PL/I STRING FUNCTIONS	69-06
360D-03.8.015	A PLAN FOR A LINKED-DIRECT FILE, WITH OPERATING ROUTINES, IN COBOL	69-13
360D-04.0.008	DOS COBOL CROSS REFERENCER	69-04
360D-04.0.010	SIMPLIFIED INPUT-OUTPUT AND DEBUGGING MACROS FOR ASSEMBLER LANGUAGE USERS	69-22
360D-04.0.009	PCMSG - PROGRAM CHECK MESSAGE	69-09
360D-04.1.010	DEBUGGING AID FOR DOS PROGRAM TESTING	69-04
360D-04.1.011	S/360 POST MORTEM DUMP (PCP/MFT)	69-09
360D-04.2.007	PROGRAM TO AID IN OPTIMIZING COMPUTE BOUND JOBS BY INDICATING THE MAJOR PROCESSING LOOPS	69-04
360D-04.6.001	A PROGRAM FOR GRAPHIC ON-LINE DEBUGGING ON AN IBM 2250 DISPLAY	69-08

360D-05.0.003	OS MASTER SCHEDULER INITIALIZATION IMPROVEMENTS	69-25
360D-05.1.016	CONVERSATIONAL REMOTE BATCH ENTRY (CRBE)	69-13
360D-05.1.018	BAYLOR EXECUTIVE SYSTEM FOR TELEPROCESSING (BEST)	69-22
360D-05.2.010	INTERACTIVE APPLICATIONS SUPERVISOR (IAS/360)	69-06
360D-05.2.011	DOS/360 TIME ACCOUNTING SYSTEMS	69-10
360D-05.5.001	44 PS LOADER	69-20
360D-06.1.005	MARC II SORT PROGRAM	69-16
360D-06.3.006	GENERALIZED TELEPROCESSING PROGRAM FOR SYSTEM/360	69-07
360D-06.3.007	DOS BTAM FOR LEASED LINE KSR 35 TELETYPES	69-06
360D-06.3.008	AUTODIN ACCESS METHOD FOR OS/360	69-10
360D-06.3.012	A HIGH SPEED BISYNCHRONOUS COMMUNICATIONS ACCESS METHOD	69-22
360D-06.5.006	UNIVAC-1108 TO IBM-360 FLOATING POINT INTERNAL CONVERTER ('CVFL08')	69-06
360D-06.5.007	7090 TO 360 FLOATING POINT INTERNAL CONVERTER ('CVFL09')	69-06
360D-06.6.003	FORTRAN CHARACTER STRING PACKAGE	69-03
360D-06.6.004	CHARACTER FILTER PL1	69-03
360D-06.7.018	BSEARCH - A RANDOM ACCESS BINARY-SEARCH TECHNIQUE FOR SEQUENTIAL FILES ON DISK OR DRUM	69-03
360D-06.7.019	KWADE-KEYWORD AS A DICTIONARY ENTRY	69-03
360D-06.7.021	AUTORAF - A SYSTEM FOR BUILDING AND USING RECORD ADDRESS FILES FOR DIRECT PROCESSING OF FILES ON SYSTEM/360	69-10
360D-06.7.024	GENERATION DATA GROUP (GDG) MODIFICATION	69-10
360D-06.8.002	LPI	69-24
360D-07.6.001	2741 READ AND WRITE ROUTINE	69-06
360D-08.0.003	WRIMAT MATRIX WRITER	69-05
360D-10.0.001	PROGRAM TO CALCULATE THE AMOUNT OF CPU TIME AVAILABLE IN THE BACKGROUND WHEN RUNNING DOS MULTIPROGRAMMING	69-03
360D-10.0.002	FORTRAN ANALYSIS PROGRAM	69-04
360D-10.2.004	HIGH WATER MARK (HWM)	69-09
360D-10.2.005	CORE AND/OR RESOURCE CONFLICT (CRC) NOTIFICATIONS	69-10
360D-10.2.006	NUCLEUS INITIALIZATION PROGRAM (NIP) MODIFICATIONS	69-11
360D-11.1.023	DISK RESIDENCE FOR 1410 COMPATIBILITY MODE PROGRAMS ON THE IBM S/360 MODEL 40	69-10
360D-12.1.020	PL/I BINARY TO EBCDIC CODE KEEPING FIVE BINARY CODED COLUMNS PRINTING RECODED DATA	69-11
360D-12.2.008	CDC - 3800/3300 COBOL TO IBM - S/360 COBOL TRANSLATOR	69-09
360D-12.2.009	NCR 315 TRANSLATOR FROM NEAT TO 360 ALP	69-09
360D-13.0.004	GRAPHICS STEPWISE REGRESSION ROUTINE (BMDGR1)	69-15
360D-13.0.005	NON-LINEAR LEAST SQUARES ON GRAPHICS (BMDGR2)	69-16
360D-13.0.006	TIME SERIES SPECTRUM ESTIMATION (BMDGT1)	69-15
360D-13.0.008	CAI DOS COURSEWRITER III CALCULATOR	69-21
360D-13.0.009	HISTOGRAM SUBROUTINE	69-16
360D-13.1.005	GENERAL TAXONOMY PROGRAM	69-08
360D-13.2.003	LEAST-SQUARES ESTIMATION OF NON-LINEAR PARAMETERS - NLIN	69-09
360D-13.4.007	RESCALED RANGE TRANSFORM	69-06
360D-13.4.008	COLLEY-TUKEY FAST FOURIER TRANSFORM -- FOUR 2 IN ASSEMBLER LANGUAGE	69-17
360D-13.6.007	NON-LINEAR LEAST-SQUARES CURVE-FITTING PROGRAM	69-20
360D-13.6.008	LINEAR LEAST-SQUARES CURVE FITTING PROGRAM	69-20
360D-13.7.001	DIALL - GENERAL LEAST SQUARES DIALLEL ANALYSIS OF VARIANCE	69-15
360D-15.0.004	OS/360 CHECKPOINT AND IPL RESTART	69-23
360D-15.4.006	ICES/360 SOURCE PROJECT-I	69-05
360D-16.0.001	UCARDS: UNION CARBIDE AUTOMATIC ROUTINE AND DESIGN SYSTEM FOR PRINTED CIRCUIT BOARDS	69-15
360D-16.2.018	BULKHEAD SHEET PILING ANALYSIS	69-06
360D-16.2.019	ICES/360 STRUDL-II	69-09
360D-16.2.020	COLUMN DESIGN - WORKING STRESS METHOD	69-09
360D-16.2.021	ICES/360 SOURCE TRANSET I	69-14

360D-16.2.022	GC-21	69-19
360D-16.2.023	ICES/360 COGO-I SOURCE	69-16
360D-16.2.024	ICES/360 ROADS I SOURCE	69-16
360D-16.2.025	ICES/360 TABLE-II, OBJECT AND SOURCE	69-16
360D-16.2.026	ICES/LEASE I	69-23
360D-16.3.001	ECONOMIC EVALUATION AND SCREENING PROGRAM FOR RESEARCH PROJECTS	69-05
360D-16.4.009	LISA 360 A PROGRAM FOR LINEAR SYSTEMS ANALYSIS	69-09
360D-16.4.010	SYNTH1 (A COMPUTER PROGRAM FOR THE DESIGN OF REACTIVE NETWORKS FROM A SET OF ARBITRARY POLES AND ZEROES)	69-13
360D-16.8.002	COMPUTER CALCULATION OF FREQUENCY RESPONSE FUNCTIONS FROM STEP RESPONSE DATA	69-19
360D-17.2.009	ON-LINE SUMX	69-23
360D-17.7.001	SUBROUTINE UCONIC	69-07
360D-19.5.008	MVT ACCOUNTING ROUTINE (MVTACTRT)	69-12
360D-19.7.010	PROOF OF DEPOSIT ENTRY RUN WITH TRANSIT DISTRIBUTION	69-08
360D-19.7.011	PERSONAL TRUST ACCOUNTING PACKAGE TO ASSIST USERS IN INSTALLING OFF-LINE PROCESSING	69-07
360D-23.1.003	TWO-STAGE, TWO-DIMENSIONAL TRIM PROGRAM II	69-10
360D-25.0.004	EQUIPMENT ORDER, SHIPMENT, INSTALLATION, AND DISCONTINUANCE REPORT FOR LARGE ACCOUNTS	69-08
360D-25.2.004	UNIT CONTROL SYSTEM FOR RETAILERS	69-08
360D-29.3.002	VSP SCHEDULE PRODUCTION FOR O/S 360	69-16
360D-30.1.002	360/2260 FUNCTIONAL DEMONSTRATION IN DOS F1 PARTITION - BTAM LOCAL	69-09
360D-30.2.002	SHARED HOSPITAL ACCOUNTING SYSTEM (SHAS) DEMONSTRATION PACKAGE	69-09
360D-40.4.003	MULTIPLE - PRECISION FLOATING-POINT ARITHMETIC PACKAGE	69-11
360D-42.2.001	EXPERIMENTAL PROGRAM FOR DETERMINING POLYNOMIAL ZEROS	69-06
360D-45.1.001	INVRSOL	69-03
360D-45.2.001	EIGENRS	69-03
360D-99.0.004	/360 DOS INDIVIDUALIZED HOMEWORK ASSIGNMENT AND QUIZ GENERATOR (QUIZ AND EXERCISE DEMONSTRATOR: QED)	69-05
360E-00.0.008	SIMPLIFIED USER ORIENTED REPORT - WRITING PROGRAM FOR 8K MODEL 20	69-05
360E-01.0.001	INVALID CCCP/360-20 INVALID CODE CARD CORRECTION PROGRAM	69-07
360E-01.0.002	FIELD SELECTION AND LIST	69-08
360E-01.5.001	MODEL 20 STAND-ALONE DISK COPY UTILITY	69-07
360E-03.4.002	SHORT INPUT OUTPUT SYSTEM	69-06
360E-06.5.002	JULIAN DATE PROCESSING SUBROUTINES	69-08
360E-06.6.003	CONVERTING A NUMBER TO IT'S VERBAL DESCRIPTION (CKPR)	69-16
360E-13.0.001	HOSPITAL PBAR PROGRAMS FOR A MFCM MODEL 20	69-07
360E-13.0.002	STATISTICAL COMPARISONS PROGRAM	69-07
360E-13.1.006	SINGLE DIGIT COUNT AND FREQUENCY DISTRIBUTIONS	69-12
360E-13.6.001	LINEAR REGRESSION IN TWO VARIABLES (LEAST SQUARE FIT) FOR THE MOD 20	69-06
360E-16.4.004	MINIMIZATION OF BOOLEAN EXPRESSIONS	69-16
360E-16.5.001	CALCULATE MASTER CAMS	69-11
360E-19.7.020	SAVINGS ACCOUNTING FOR THE 360 MOD 20 DISK	69-04
360E-23.1.003	HEURISTIC PAPER MACHINE TRIM PROGRAM FOR THE IBM 360/20	69-10
360E-25.2.006	DOPTIM - DISK ORDER POINT TECHNIQUE OF INVENTORY MANAGEMENT	69-18
360E-27.3.001	AMERICAN FLYERS AIRLINE AVAILABILITY REPORT	69-13
360E-28.3.013	360/20 CARD DESCRIPTIVE BILLING	69-09
360E-28.3.014	WHOLESALE AND/OR RETAIL, ACCOUNTS RECEIVABLE, AGED TRIAL BALANCE	69-06
1130-00.0.009	SFREF - FORTRAN AND ASSEMBLY LANGUAGE SUBPROGRAM,	

	LINK, AND FILE READ/WRITE CROSS-REFERENCE TABLE BUILDING PROGRAM	69-11
1130-00.0.011	CDUTL, A GENERAL PURPOSE CARD UTILITY	69-25
1130-00.3.010	1130 PRNZ MODIFIED FOR ADDITIONAL CARRIAGE CONTROL	69-07
1130-00.4.007	1130 DISK-PACK SECURITY PROGRAM	69-16
1130-01.5.006	DUMP AND RESTORE MONITOR VERSION TWO	69-21
1130-01.6.003	ISFMS - 1130 INDEXED SEQUENTIAL FILE MANAGEMENT SYSTEM ON THE 1130	69-05
1130-03.4.011	REREAD FOR FREE FORMAT CARD READING IN FORTRAN	69-10
1130-05.5.001	SECAL - A PROGRAM FOR BUILDING MULTIPLE LOCAL SUBROUTINE MODULES FOR THE IBM 1130	69-10
1130-06.1.006	UDAC 1130 DSORT	69-20
1130-06.1.007	ORDER. AN IN-CORE TAG SORT	69-23
1130-06.3.001	IBM 1130 REMOTE JOB ENTRY SYSTEM	69-06
1130-06.3.002	WILLIAM AND MARY REMOTE JOB ENTRY PROGRAM - 8K DISK SYSTEM VERSION (WMRJ8)	69-11
1130-06.3.003	WILLIAM AND MARY REMOTE JOB ENTRY PROGRAM - 4K DISK AND CARD SYSTEMS VERSION	69-11
1130-06.6.014	THE 1130/1800 KEYWORD INDEX GENERATOR	69-04
1130-06.6.015	IBM 1130 FORTRAN REREAD DEVICE AND FREE FORMAT KEYWORD INPUT ROUTINES	69-06
1130-06.6.016	ALPHAMERIC DATA STATEMENT PRODUCER FOR FORTRAN	69-09
1130-06.6.017	KWIC INDEX GENERATOR	69-19
1130-06.7.003	IBM 1130 INDEX FOR DM2 SYSTEMS	69-11
1130-07.0.002	RASEC/RDSEC	69-11
1130-08.0.002	HIGH SPEED AND OVERLAP 1132 PRINTER ROUTINES FOR 1130 FORTRAN AND BAL USERS	69-16
1130-12.1.005	1130 FILE CONVERSION PROGRAM, DISK MONITOR VERSION I TO DISK MONITOR VERSION 2 (FCOPY)	69-10
1130-13.7.001	MULTIPLE RANGE TEST FOR CORRELATED AND HETEROSCEDASTIC MEANS	69-08
1130-15.1.002	GASP II (GENERAL ACTIVITY SIMULATION PROGRAM, VERSION II)	69-23
1130-15.2.005	1130 BLENDING OF INITIAL FURNACE CHARGES	69-10
1130-15.4.007	W.E.L. CHART SCHEDULING PROGRAM	69-16
1130-15.5.001	PURDUE FARM SUPPLY CENTER MANAGEMENT GAME	69-19
1130-15.5.002	PURDUE SUPERMARKET MANAGEMENT GAME, 1969	69-20
1130-16.5.002	MECHANICAL TOLERANCE ANALYSIS PROGRAM (MTAP)	69-22
1130-17.3.002	GENERAL PURPOSE CHROMATOGRAPH PEAK INTEGRATION PROGRAM	69-07
1130-17.3.004	FAST	69-25
1130-17.6.002	1130 RAW SCORE TEST ANALYSIS PROGRAM	69-15
1130-17.6.003	1130 TEST ITEM ANALYSIS PROGRAM	69-16
1130-17.6.004	APGRD A PROGRAM TO GRADE AND ANALYZE MULTIPLE CHOICE QUIZZES	69-18
1130-21.1.001	IBM 1130 HOSPITAL PAYROLL	69-09
1130-23.3.002	COMPUTER APPROACH TO INVENTORY REQUIREMENTS ON THE 1130 FOR SMALL ASSEMBLY ORIENTED MANUFACTURERS	69-09
1130-28.2.002	1130 PATIENT BILLING	69-08
1130-28.3.001	1130 HOSPITAL ACCOUNTS RECEIVABLE	69-08
1130-30.1.004	INDIVIDUALIZED HOMEWORK ASSIGNMENT AND QUIZ GENERATOR (QUIZ AND EXERCISE DEMONSTRATOR: Q.E.D.)	69-04
1130-30.2.003	BATTLE OF THE NUMBERS	69-12
1130-30.2.004	1130 PERPETUAL CALENDAR DEMONSTRATION PROGRAM	69-13
1130-30.2.005	MOON FLIGHT	69-19
1130-45.0.001	SIMPLIFIED GAUSS-JORDAN METHOD OF MATRIX INVERSION	69-20
1500-01.6.001	RAS/1130 COURSEWRITER II REASSEMBLE PROGRAM	69-24
1500-99.0.001	FUNCTION SF FOR ANY 1500 COURSEWRITER II CAI SYSTEM	69-20
1500-99.0.002	FUNCTION EP 1500 CW II FUNCTION CAI SYSTEM	69-24
1800-00.0.002	OBJECT TIME I/O LOGICAL NUMBER GENERATOR	69-04
1800-00.0.003	CARD UTILITY PROGRAM USING 1800-OS-010, MPX	69-10
1800-01.0.002	BOM CORE & DISK DUMP UTILITIES MODIFIED TO RUN IN A CORE LOAD AREA OF V-CORE FOR USE WITH 1132 OR 1443 LINE PRINTERS	69-11

1800-17.4.001	REDUCTION OF OCEANOGRAPHIC STATION DATA	69-09
1800-23.5.006	IBM ANALOG DATA ACQUISITION PROGRAM	69-04
1800-23.5.008	PROCESS DATA PREPARATION PROGRAM	69-10
1800-25.2.001	COMPUTER PROGRAM FOR A HOSPITAL BLOOD BANK	69-19
1401-10.3.071	1401 UNIVERSITY ADMISSIONS INFORMATION SYSTEM	69-14
1401-10.3.072	FINANCIAL AID INFORMATION SYSTEM (LOAN SUBSYSTEM)	69-14
1401-10.3.073	FINANCIAL AID INFORMATION SYSTEM (STUDENT AID SUBSYSTEM)	69-14
1620-01.6.160	LABELR - A REPORT GENERATOR	69-15
1620-02.0.066	A MODIFIED PDQ FORTRAN SYSTEM	69-16
7040-H2 3552	MULTIDIMENSIONAL OPTIMIZATION USING PATTERN SEARCH	69-11
7094-F2 3554	EIGENVALUE - EIGENVECTOR ROUTINE FOR REAL COMPLEX MATRIX	69-15

IBM Installation Newsletter

April 24, 1970 Issue No. 70-08

CONTENTS

(G means Green Section)

Distribution: By DAPS Subscription.
Order Number 0925.

*Requires Immediate Attention

ZZ20-7008 Printed in U.S.A.

SELECTED SHORT SUBJECTS

The purpose of Selected Short Subjects is to bring together and highlight concise, factual and timely information indicating that action is to be taken by the IBM representative whose accounts are affected.

1. IBM PROGRAM PRODUCT PROTECTION

IBM Program Products and the documents describing them are protected by copyright laws. A booklet titled "IBM Program Products, Instructions on Copyright Notice" (G120-2083) discusses the main provisions of the License Agreement as they relate to the copyright laws. Copies of this booklet are now available for your customers from the IBM Distribution Center, Mechanicsburg. This booklet helps in understanding the use of copyrighting as a method of program protection. A copy of this booklet is included with each Program Product shipped from PID. All sales representatives, system engineers and field engineers should be aware of the protection requirements outlined in this booklet and the terms and conditions of the recently revised License Agreement for IBM Program Products (Marketing Announcement 270-22, dated 3/2/70). Care must be taken not to initiate any action in conflict with the protection requirements for Program Products.

2. NEW RELEASES OF PROGRAMS

This information is furnished to ensure that you are aware of new releases of programs which are being used by your customers. The programming service period for program releases being replaced by new releases is stated below.

1. CPS (Conversational Programming System) Version 5 Modification Level 0, 360D-03.4.016 is now available from PID. CPS Version 5 Modification Level 0 has a Programming Service Classification of A. CPS Version 5 Modification Level 0 will remain "current" for purposes of programming service until 3 months after the date of the next CPS release. For programming service purposes, CPS Version 4 will be considered "current" until May 27, 1970.

2. HASP-II Version 2 Modification Level 3 (360D-05.1.014) is now available from PID. HASP-II has a Programming Service Classification of A. Version 2 Modification Level 3 will remain "current" for purposes of programming service until the OS/360 release with which it operates is no longer current. All prior releases of HASP-II Version 2 will remain "current" for purposes of programming service as long as OS/360 Release 17 remains current. PID has sent ordering instructions and information to registered users of HASP-II.

3. POWER II - Version 2 Modification Level 0 (360D-05.2.006) is now available from PID. POWER II - Version 2 Modification Level 0 has a Programming Service Classification of A. For programming service purposes, POWER II Version 1 will be considered "current" for the same period as Release 21 of DOS. If in the future, a new release is made available for this program, the period that Version 2 Modification Level 0 remains current will be specified at the time of the new release. POWER II Version 2 Modification Level 0 contains maintenance and the RJE capability has been added. For further information on POWER II see Branch Managers Memorandum Number B70-60.

4. S/360 Project Control System (360A-CP-06X) Version 2 Modification Level 3 is available from PID. Version 2 Modification Level 2 will remain "current" for programming Service Purposes until July 1, 1970.

3. INL DROPPING PRL REPRINTS FINAL NOTICE

Since the weekly Publications Release Letter (PRL) is available to each of you by subscription under the expanded DAPS system, the reprinting of the New Publications List will be discontinued after this issue. Advance notice of this planned action was published in two recent INLs.

Those of you who want to continue to receive this list should subscribe to the PRL, Letter Only, DAPS Order No. 0076.

OS/360 ALTERNATE LINKLISTS AT IPL

This contribution has not been submitted to any formal IBM test. Potential users should evaluate its usefulness in their own environment prior to implementation.

Shown in Exhibit 1 is an updated listing to the IEAANIP macro in SYS1.MODGEN for Release 18. This modification permits the reading of an alternate linklist at IPL time.

The operator now has the option of reading one of ten different linklists (LNKLST00-LNKLST09). The desired list is specified at "SPECIFY SYSTEM PARAMETER" time by replying with the key-word r00, 'LNK=x', x being a number 0 through 9, and 0 being default.

This facility is easily incorporated into an existing Release 18 system by reassembling IEANIP0 with the same parameters as during Stage 2 of the System Generation and then relinking the nucleus as shown in Exhibit 2.

The user will thus have the added flexibility in his system of not having to update SYS1.PARMLIB each time his list of concatenated link libraries changes.

```
./ CHANGE NAME=IEAANIP,LIST=ALL
         BNE   JPLNKLS  NO,TEST FOR NEXT PARAMETER                          15736018
JPLNKLS  CLC   0(4,6),JPLNK+1  IS THIS PARM 'LNK= '                         15780019
         BNE   TMSLPARM      BRANCH IF NO - CHECK FOR NEXT PARM             15780020
         LA    2,IEABLANK                                                   15780021
         MVI   EOBTEST+1,X'F0'   NO 'U' THE SECOND TIME AROUND              15780022
         LA    6,3(6)         BUMP POINTER TO PARM VALUE                    15780023
         LA    6,1(6)       GET PARM VALUE                                  15780024
         CLI   0(6),C'9'      CHECK FOR VALID PARAMETER                     15780025
         BH    IEAOPERR         BRANCH IF THE                               15780026
         CLI   0(6),C'0'          PARAMETER IS NOT                          15780027
         BL    IEAOPERR       NUMERIC -- I.E. PRINT ERROR MESSAGE           15780028
         MVC   IEALNKLS+7(1),0(6)  CHANGE NAME OF LNKLST00 TO NEW NAME      15780029
         LA    6,1(6)       BUMP POINTER AND                                15780030
         CLI   0(6),C','              PREPARE                               15780031
         BNE   IEABLANK             FOR RETURN -  TO CHECK                  15780032
         LA    6,1(6)         FOR THE NEXT                                  15780033
         B     IEABLANK           PARAMETER                                 15780034
JPLNK    DC    C' LNK= '      KEYWORD PARM TO CHANGE LNKLST00               69150016
./ ENDUP
```

EXHIBIT 1

```
//LINKNUC JOB (1370,72),POHLMANN,MSGLEVEL=1,REGION=200K
//LKED EXEC PGM=IEWL,PARM='NCAL,LIST,XREF,SCTR,LET,DC'
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD UNIT=SYSDA,SPACE=(CYL,(20,20))
//IEANUCOX DD DSN=SYS1.NUCLEUS,DISP=OLD,UNIT=2314,VOL=SER=MVTIPL
//SYSLMOD DD DSN=JPNUC,UNIT=2314,SPACE=(CYL,(9,9,2)),DCB=SYS1.NUCLEUS,
//        DISP=(,PASS)
//SYSLIN DD *
          .
          .
          .
   IEANIP0 OBJECT DECK
          .
          .
          .
 INCLUDE IEANUCOX(IEANUC01)
 NAME IEANUCOJ
 /*
//STWO EXEC PGM=IEWL,PARM='NCAL,LIST,XREF,SCTR,LET,DC'
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD UNIT=SYSDA,SPACE=(CYL,(20,20))
//SYSLMOD DD DSN=SYS1.NUCLEUS,DISP=SHR
//NUCLEUS DD DSN=JPNUC,DISP=(OLD,DELETE)
//SYSLIN DD *
 INSERT IEAANIP0
 INSERT IEAQFX00
 INCLUDE NUCLEUS(IEANUCOJ)
 NAME IEANUC01(R)
 /*
```

EXHIBIT 2

For IBM Internal Use Only

OS/360 MFT ALTERNATE NUCLEUS FOR VARIED PARTITIONS

This contribution has not been submitted to any formal IBM test. Potential users should evaluate its usefulness in their own environment prior to implementation.

Under the same title in INL 69-10, a modification to the OS Release 15/16 and 17 IPLTXT was published. This modification permitted the user to select an alternate nucleus through the use of the console. This modification has been implemented under Release 18 of the Operating System; however, the following changes to the modification were necessary:

1. The address of IEABUFR had to be changed to 2912.

2. Instructions have been added to change lower case typewriter input to upper case, and thus eliminate the need of a second alias for a nucleus with an alphabetic identifier.

Exhibit 1 is a copy of the source listing for the IPLTXT modification for OS Release 18.

EXHIBIT 1

```
STMT   SOURCE STATEMENT                                                       F01FEB69   1/12

ZERO      START 0                                                             IPL001
*                                                                             IPL001
*         THESE INSTRUCTIONS SIMULATE THE BASIC FUNCTIONS, AS DESCRIBED       IPL001
*         IN THE OPERATOR'S GUIDE, OF SELECTING AN ALTERNATE NUCLEUS.         IPL001
*         ANY CHARACTER MAY BE ENTERED AS LONG AS IT MATCHES THE              IPL001
*         "IEANUCO*" * IDENTIFYING CHARACTER.                                 IPL001
*         USING ALIAS, YOU MIGHT HAVE MORE MEANINGFUL IDENTIFICATION.         IPL001
*         EXAMPLE  D FOR DAY SHIFT, C FOR COMMERCIAL ETC.                     IPL001
*         OR EVEN 1 FOR IEANUC01, 2 FOR IEANUC02, ETC.                        IPL001
*         NOTE  UPPER OR LOWER CASE MAY BE ENTERED FROM THE CONSOLE           IPL001
*                                                                             IPL001
          USING *,0                                                           IPL001
          ORG   ZERO+72                 CAW ADDRESS                           IPL001
          DC    AL4(CCWS)               ADDRESS OF CCW'S                      IPL001
          ORG   ZERO+130                ADDRESS OF INSTR AFTER BALR IN IPLTX  IPL001
          B     PATCHES                 OVERLAY LM INSTR W/BRANCH TO PATCHES  IPL001
          ORG   ZERO+3120               ADDRESS BEYOND IPLTXT                 IPL001
PATCHES   SIO   CONSOLE                 TYPE IDENTIFICATION MESSAGE           IPL001
          BC    7,PATCHES               AND                                   IPL001
TIO       TIO   CONSOLE                 ASK FOR DESIRED NUCLEUS               IPL001
          BC    7,TIO                   MAKE SURE THAT IO COMPLETE            IPL001
          XC    CSW(12,0),CSW(0)          ZERO THE CAW AND CSW BEFORE RET     IPL001
          CLI   LOC8,X'00'              HAS ANYTHING BEEN TYPED IN
          BE    NOUPCASE                BRANCH IF NO
          OI    LOC8,C' '               MAKE ALPHA TO UPPER CASE              IPL001
NOUPCASE  EQU   *
          LM    0,14,IEABUFR            EXECUTE THE OVERLAID INSTR            IPL001
          B     CONTINUE                BRANCH TO INSTR AFTER (B PATCHES)     IPL001
*                                                                             IPL001

CCWS      CCW   9,IDENT,X'60',22        TYPE IDENT, CARR RETURN               IPL001
          CCW   9,IDENT2,X'60',34       AND                                   IPL001
          CCW   1,ASK,X'60',27          TYPE QUESTION, NO CARR RETURN         IPL001
          CCW   10,LOC8,X'20',1         READ ONE CHAR TO IDENTIFY NUC MEMBER  IPL001
IDENT     DC    CL22' OPERATING SYSTEM /360'                                  IPL001
IDENT2    DC    CL34' ENTER DIGIT FOR DESIRED NUCLEUS     '                   IPL001
ASK       DC    CL27' OR HIT EOB FOR DEFAULT - '     IEANUC01                 IPL001
LOC8      EQU   8                       LOCATION 8 IN CORE                    IPL001
CONSOLE   EQU   X'1F'                   CONSOLE ADDRESS                       IPL001
CSW       EQU   64                      CSW ADDRESS                           IPL001
CONTINUE  EQU   134                     ADDR OF INSTR TO RETURN CONTROL       IPL001
*                                                                             IPL001
IEABUFR   EQU   2912   (R-18)           R15 AFTER BALR + IEABUFR IN IPLTXT    IPL001
*         EQU   2848   (R-17)           R15 AFTER BALR + IEABUFR IN IPLTXT    IPL001
*         EQU   2800   (R-15/16)        R15 AFTER BALR + IEABUFR IN IPLTXT    IPL001
          END                                                                 IPL001
```

DOS/360 CATALOGING TECHNIQUE FOR BILL OF MATERIAL PROCESSOR

This contribution is an improvement over the technique described in INL 68-20. It has not been submitted to any formal IBM test. Potential users should evaluate its usefulness in their own environment prior to implementation.

When cataloging a load module to the Core Image Library with the same phase name as one already in the library, only the latest phase is accessible to DOS. This applies to Bill of Material Processor phases for I/O Open/Close, I/O Process and DEARUX when several programs of a system use these same modules.

While the approaches described in INL 68-20 will all work, a better approach is as follows:

1. Place the Include card for the I/O Root phase immediately behind the Phase card.

2. Follow that card with the Include cards for any interface modules as required by the programming language used.

3. Then include the object module for the user program.

4. Finally, put in the Include cards for the I/O Open/Close and I/O Process modules.

See the sample Link Editor listing in Exhibit 1.

The user is cautioned that if any of the Bill of Material Processor customizing or maintenance programs (such as DEACUC, DEACUL, DEACUM, etc.) are cataloged, they use DEARUX and it will be likely to have a load address different from its address in user programs. Therefore, these programs should be run in a link-and-execute mode rather than being cataloged.

```
// JOB CATALOG BILL OF MATERIAL PROCESSOR USER PROGRAMS                00.00.04
// OPTION CATAL
   PHASE SLRET,*
   INCLUDE DEARS7        I/O ROOT MODULE
   INCLUDE DEART1        COBOL INTERFACE MODULE
   INCLUDE DEART2        COBOL CHASE MODULE
   INCLUDE DEART3  USER PROGRAM-COMPILE OR OBJECT DECK
   INCLUDE DEARS8        I/O PROCESS MODULE
   INCLUDE DEARS9        I/O OPEN-CLOSE MODULE
   ENTRY
// LBLTYP NSD(3)
// EXEC LNKEDT
```

```
JOB  CATALOG   04/01/70   DISK LINKAGE EDITOR DIAGNOSTIC OF INPUT

ACTION TAKEN    MAP
LIST      PHASE SLRET,*
LIST      INCLUDE DEARS7        I/O ROOT MODULE
LIST      INCLUDE DEART1        COBOL INTERFACE MODULE
LIST      INCLUDE DEART2        COBOL CHASE MODULE
LIST      INCLUDE DEART3  USER PROGRAM-COMPILE OR OBJECT DECK
LIST      INCLUDE DEARS8        I/O PROCESS MODULE
LIST   AUTOLINK   IJDFAPZZ
LIST            PHASE DEARS8,XZ$OVLY,NOAUTO                          BM ZA038
LIST      INCLUDE DEARS9        I/O OPEN-CLOSE MODULE
LIST          INCLUDE DEARUX                                         BMAYA01P
LIST            PHASE DEARUX,XZ$OVLY,NOAUTO                          BM CX001
LIST            PHASE DEARS9,XZ$OVLY,NOAUTO                          BM YA015
LIST      ENTRY
```

EXHIBIT 1

DOS/360 CATALOGING TECHNIQUE FOR BILL OF MATERIAL PROCESSOR

04/01/70	PHASE	XFR-AD	LOCORE	HICORE	DSK-AD	ESD TYPE	LABEL	LOADED	REL-FR
	SLRET	005720	003890	006D68	40 05 2	CSECT	DEARS7	003890	003890
						ENTRY	BM$PIO	003890	
						ENTRY	XA$RET1	0038F8	
						ENTRY	MS$GRT	003910	
						ENTRY	XL$COMM	003988	
						* ENTRY	X7$OVLY	003FA8	
						CSECT	DEART1	0048A8	0048A8
						CSECT	BMPOPEN	004880	0048A8
						CSECT	BMPCLOSE	004BE0	0048A8
						CSECT	BMPCALL	004C40	0048A8
						CSECT	BMPGET	004CC8	0048A8
						CSECT	BMPPUT	004CF0	0048A8
						CSECT	BMPSTKY	004D18	0048A8
						CSECT	BMPSTDA	004D40	0048A8
						CSECT	BMPEOF	004D68	0048A8
						CSECT	DEART2	004DA0	004DA0
						CSECT	CHASE1	004EE8	004DA0
						ENTRY	CHASER1	004EFA	
						CSECT	CHASE2	004F20	004DA0
						ENTRY	CHASER2	004F32	
						CSECT	CHASE3	004F58	004DA0
						ENTRY	CHASER3	004F6A	
						CSECT	CHASE4	004F90	004DA0
						ENTRY	CHASER4	004FA2	
						CSECT	DEART3	004FC8	004FC8
						CSECT	IJDFAPZZ	006C60	006C60
						* ENTRY	IJDFAZZZ	006C60	
	DEARS8	003FA8	003FA8	0047F7	40 09 2	CSECT	DEARS8	003FA8	003FA8
						ENTRY	YA$RT	003FA8	
	DEARUX	003FA8	003FA8	0041E3	41 00 2	CSECT	DEARUX	003FA8	003FA8
	DEARS9	003FA8	003FA8	00474F	41 01 1	CSECT	DEARS9	003FA8	003FA8

EOJ CATALOG 00.00.47,DURATION 00.00.43

EXHIBIT 1 cont'd.

INFORMATION ABOUT THE NEWSLETTER

The IBM Installation Newsletter is distributed only to IBM locations and is not available to customers. The Newsletter is separated into two parts. The first part contains information which is intended only for IBM personnel and the pages are labeled For IBM Internal Use Only. THESE PAGES, OR REPRODUCTIONS OF THEM, ARE NOT TO BE GIVEN TO CUSTOMERS. However, using your own good judgement, you may discuss pertinent information from these pages with a customer. The second part contains information which may be reproduced by the Branch Office at their discretion and given to customers. These pages are not labeled with any restrictive classification. Note that when any abstracts for Type III and Type IV programs are given to the customer, he must be informed that the disclaimers for Type III and Type IV programs as contained in the Newsletter, applies.
YOUR COOPERATION IN PROPERLY HANDLING THE CONTENT OF THE IBM INSTALLATION NEWSLETTER IS NECESSARY FOR ITS CONTINUED PUBLICATION.

Direct inquiries concerning the contents of this Newsletter to:

W. F. Malmborg, Editor
IBM Installation Newsletter
Dept. 812, DPD HQ
112 East Post Road
White Plains, New York 10601
Telephone 914-696-3781
Tie Line 254-3781

NEW PUBLICATIONS ANNOUNCED BY PRLs

Information is extracted and condensed from the weekly PRLs (Publications Release Letters) to insure that all Salesmen and Systems Engineers are aware of new or revised Reference Sources, Marketing Publications, and Field Engineering Publications. Normally, each issue of the Newsletter will contain information extracted and condensed from two PRLs, one following the other. The information will be placed in the Newsletter in its original sequence with no rearrangement of form numbers or titles. It is not intended to replace existing information and distribution sources. You should be certain that you are aware of these sources.

Do not use the condensed Newsletter information to order publications. Most but not all (for example, some films and slide sets) of the items are available from Mechanicsburg. To receive the items you need without delay, prior to ordering check the REQ (Requisition Form) code in the "Accumulative Supplement to IBM Publications Current Price List (Z20-0100)" which is attached to the weekly Publications Release Letter (PRL) and distributed to your Administration Manager. For an explanation of the REQ codes and other publications information see the first few pages of the "IBM Publications Current Price List Z20-0100".

The IBM publications requisition (MO2-0618) or ITPS should be used for ordering items from the IBM Distribution Center, Mechanicsburg, Pa.

(IBM World Trade Corporation has its own distribution center and procedures.)

PRL #12 March 27, 1970

ORDER NUMBER	TITLE/DESCRIPTION	DISPOSITION	ABST#	CLASS	DAPS CODE	SL/ SS	BOL QTY
	MARKETING PUBLICATIONS						
GA24-3021-5	IBM 1030 DATA COLLECTION SYSTEM INSTALLATION MANUAL - PHYSICAL PLANNING	SCRAP**	04002	SRL	0489	YES	ONE
GA24-3509-1	IBM 2944 DATA CHANNEL REPEATER INSTALLATION MANUAL - PHYSICAL PLANNING	SCRAP**	04513	SRL	0489	YES	ONE
GA26-5921-5	IBM 1800 BIBLIOGRAPHY	SCRAP**	03777	SRL	0006	YES	ONE
GC26-3706-5	IBM 1130 SYNCHRONOUS COMMUNICATIONS, ADAPTER SUBROUTINES	SCRAP**	05529	SRL	0158	YES	ONE
GH20-0327-2	GENERAL PURPOSE SIMULATION SYSTEM/360, DISK OPERATING SYSTEM (360A-CS-19X) OPERATOR'S MANUAL	USE***	03352	APPRG		YES	
GH20-0340-3	STRUCTURAL ENGINEERING SYSTEM SOLVER (STRESS) FOR THE IBM 1130, MODEL 2B (1130-EC-03X) VERSION 2, USER'S MANUAL	USE***	03350	APPRG		YES	
GH20-0507-1	BIBLIOGRAPHY OF APPLICATION PUBLICATIONS FINANCE INDUSTRIES	SCRAP**	03961	APPRG		YES	TWO
GH20-0683-1	APL/360 USER'S MANUAL	USE***	11617	SRL		YES	
GH20-4058-0	CLINICAL LABORATORY MANAGEMENT SYSTEM 5718-H12		13748	PRGPR		NO	
GK20-0355-0	ONLINE BANKING APPLICATIONS FOR 2260 DISPLAY STATIONS AND 2740 COMMUNICATION TERMINALS		13596	BRIEF		NO	TWO
GL26-2114-0	IBM VIDEO DISPLAY WITH IBM 1800		13749	ADTV		NO	ONE
ZM02-6420-3	MOVING AND LIVING EXPENSE ACCOUNT FORM	SCRAP**	13757	FORM		NO	
GN20-1001-46	TELEPROCSING-SRL-NL-A24-3089	SCRAP**			0306	YES	ONE
GN20-1002-27	DP TECHNIQUES NL-RE-F20-8172	SCRAP**			0070	YES	ONE
GN20-5201	TNL 220-2002-00				0903	YES	ONE
GN20-5400-7	220-2004-00 TNL	USE***			0905	YES	ONE
GN20-5407	TNL 220-2004-00				0905	YES	ONE
GN20-5506	TNL 220-2005-00				0906	YES	ONE
GN33-8617	C24-9003-04 TNL				0381	YES	ONE
ZR20-1116-0	FUNDAMENTALS OF PROGRAMMING EXAMINATION		13766	TEXT		NO	
ZR20-4148-0	TERMINAL ORIENTED SYSTEMS MARKETING GUIDE		13750	ED GD		NO	
GR20-4150-0	SYSTEM/360 COMMUNICATIONS SYSTEM DESIGN AND ANALYSIS		13751	FE ED		NO	
SR20-4161-0	PRESENTATION SKILLS WORKSHOP		13752	ED GD		NO	
ZR20-6015-0	SYSTEM/3 DISK SYSTEM DESIGN EXAMINATION SCORING KEY		13753	ED GD		NO	
SR20-6020-0	SYSTEM/3 DISK SYSTEM IMPLEMENTATION, STUDENT EXERCISES		13597	ED GD		NO	
ZR20-6021-0	SYSTEM/3 DISK SYSTEM DESIGN		13754	MANUL		NO	
GR20-9163-3	SYSTEM/360 COMMUNICATIONS SYSTEM DESIGN AND ANALYSIS COURSE CODE D3602	SCRAP**	06439	FLDR		YES	ONE
GV20-6072-0	SYSTEM/3 APPLICATION CUSTOMIZER		13772	FLIP		NO	
GX20-1739-2	SYSTEM/360 REFERENCE DATA LINKAGE EDITOR	SCRAP**	04238	REFCD		YES	ONE
GX24-3808-1	0077 IBM CONTROL PANEL TEMPLATE	SCRAP**	11169	TEMPL		NO	
GYB0-0507-0	SYSTEM/360 DOS POWER II: LISTINGS SYSTEM MANUAL		13756	PROG4		NO	
GY20-0398	Y20-0211-01 TNL					YES	
ZY23-4018-0	COMMUNICATIONS SKILLS WORKSHOP		13759	ED GD		NO	
ZZ20-7005-0	INSTALLATION NEWSLETTER ISSUE 70-05 3/13/70		13782	PAPER	0925	NO	ONE
ZZ22-6947-0	IBM SYSTEMS MEASUREMENT INSTRUMENT COMPONENT DESCRIPTION AND USER'S GUIDE		13761	MANUL		NO	ONE
ZZ77-0030-0	BENCHMARK MANAGEMENT ANALYSIS PROCEDURE: A SERIES OF JOB STREAM OPTIMIZATION TOOLS		12710	PAPER		NO	
ZZ77-0032-0	SELLING AND PERFORMING SE SERVICES		12714	PAPER		NO	
ZZ77-0033-0	OS/MVT PERFORMANCE MEASUREMENT TECHNIQUES		12708	PAPER		NO	
Z120-2065-1	LICENSE AGREEMENT FOR IBM PROGRAM PRODUCTS	SCRAP**	13758	FORM		NO	
Z120-4437-28	BUSINESS RECORD FORM	SCRAP**	13783	PAD		NO	
G220-2000-0	PROGRAMMING SYSTEMS MEMORANDA (CPSM) 1130 PROGRAMMING SYSTEMS		13602	REFCD	0902	YES	ONE
G220-2001-0	PROGRAMMING SYSTEMS MEMORANDA (PSM) 1800 PROGRAMMING SYSTEMS		13603	REFCD	0901	YES	ONE
G220-2006-7	PROGRAMMING SYSTEMS MEMORANDA (PSM) SYSTEM/360: BPS PROGRAM SYMPTON INDEX	SCRAP**	11966	REFCD	0907	YES	ONE
G220-2007-7	PROGRAMMING SYSTEMS MEMORANDA (PSM) SYSTEM/360: BOS PROGRAM SYMPTON INDEX	SCRAP**	11967	REFCD	0908	YES	ONE
G220-2008-7	PROGRAMMING SYSTEMS MEMORANDA (PSM) SYSTEM/360 : TOS PROGRAM SYMPTON INDEX	SCRAP**	11968	REFCD	0909	YES	ONE
G220-2009-7	PROGRAMMING SYSTEMS MEMORANDA (PSM) SYSTEM/360: DOS PROGRAM SYMPTON INDEX	SCRAP**	11969	REFCD	0910	YES	ONE
G220-2010-7	PROGRAMMING SYSTEMS MEMORANDA (PSM) SYSTEM/360: CONVERSION PROGRAM SYMPTON INDEX	SCRAP**	11970	REFCD	0911	YES	ONE
G220-2011-7	PROGRAMMING SYSTEMS MEMORANDA (PSM) SYSTEM/360: 44PS/BPS PROGRAM SYMPTON INDEX	SCRAP**	11971	REFCD	0912	YES	ONE
G221-0512-1	IBM BASIC SYSTEMS CENTERS - PROPOSAL INSERT	USE***	10587	PROIN		NO	ONE
Z320-1046-1	BOIS KWOC INDEX (MARKETING, FIELD ENGINEERING BRANCH OFFICE ADMIN.)	SCRAP**	10821	MANUL	0077	NO	

* - REPLACED BY NEW ITEM LISTED; SEE OBSOLETE/REPLACE CATALOG.
** - SCRAP PRIOR EDITIONS.
*** - USE PRIOR EDITION.

***** IBM INTERNAL USE ONLY *****

PRL #12

ORDER NUMBER	TITLE/DESCRIPTION	DISPOSITION	ABST#	CLASS	DAPS CODE	SL/ SS	BOL QTY
G320-1060-0	IBM AGRIBUSINESS SYMPOSIUM - FEED AND GRAIN		13589	MANUL		NO	
Z502-0003-1	SUGGESTION INVESTIGATORS HANDBOOK BINDER	USE***	13747	ADTV		NO	
G502-3000-2	1969 ANNUAL REPORT	SCRAP**	13784	BROCH		NO	
	FIELD ENGINEERING						
GN20-5201	TNL 220-2002-00				0903	YES	
GN20-5400-7	220-2004-00 TNL	USE***			0905	YES	
GN20-5407	TNL 220-2004-00				0905	YES	
GN20-5506	TNL 220-2005-00				0906	YES	
SR23-3168-0	2050 PU CHANNELS AND STORAGE		13774	FEINS		NO	
SR23-3181-1	2415 MAGNETIC TAPE UNIT AND CONTROL MODELS 4, 5, AND 6	SCRAP**	12699	FE ED		NO	
SR23-3192-0	2050 PU STORAGE AND CHANNELS		13775	FEINS		NO	
SR25-5442-1	2760 OPTICAL IMAGE UNIT		13755	ED GD		NO	
SR25-5512-0	DOS ANS COBOL		13776	FE ED		NO	
SR27-9647-0	2020 BSCA SUBMOD 2 TO SUBMOD 5		13777	FE ED		NO	
SS27-0658	FES 127-0905-01					NO	
ZY25-2561-2	MLC CORPORATE MACHINE LEVEL CONTROL	USE***	04433	FE ED		NO	
ZY25-2601-1	SLT PACKAGING AND DOCUMENTATION	USE***	07437	FE ED		NO	
ZY25-3918-1	5496 DATA RECORDER QUIZ #1	SCRAP**	13778	FE ED		NO	
ZY25-3919-1	5496 DATA RECORDER QUIZ #2	SCRAP**	13779	FE ED		NO	
ZY25-3920-1	5496 DATA RECORDER QUIZ #3	SCRAP**	13780	FE ED		NO	
ZY25-3981-0	5486 CARD SORTER STUDENT QUIZ		13760	FE ED		NO	
SY26-4129-2	1800 DATA ACQUISITION AND CONTROL SYSTEM VOLUME 2 (FV THROUGH YD)	USE***	04129	FEMDM		NO	
SY26-4131-1	2310 DISK STORAGE, MODELS A1, A2, A3	USE***	04127	FEMDM		NO	
SY27-0047-0	2972 MODEL 8 GENERAL BANKING TERMINAL SYSTEM		13770	FETOM		NO	
SY33-1028-1	2520 CARD READER PUNCH ATTACHMENT FEATURE SYSTEM/360 MODEL 20 (MACHINES WITH SERIAL NO. 50,000 AND ABOVE)	USE***	11548	FEMDM		NO	
ZZ20-7005-0	INSTALLATION NEWSLETTER ISSUE 70-05 3/13/70		13782	PAPER	0925	NO	
ZZ25-2109-6	CD PROJECT INDENT PAGE	USE***	11937	FORM		NO	
Z120-1899-2	INVENTORY ERROR (CORRECTION REQUEST)	USE***	13763	FORM		NO	
S127-0907-2	MICROFICHE - ILLUSTRATED PARTS CATALOG 2844 AUXILIARY STORAGE CONTROL	SCRAP**	13685	IPC		NO	
S131-0021-0	2970 BANK TERMINAL (MODEL 8)		13764	IPC		NO	
S2C0-0050-17	MICROFICHE - 2025 AND LARGER NEW CARD	SCRAP**	12857	FEEW		NO	
S2C0-0100-17	MICROFICHE - OPERATING SYS OS NEW CARD	SCRAP**	12858	FEEW		NO	
S2C0-0149-13	MICROFICHE - OPERATING SYS OS NEW CARD	SCRAP**	12859	FEEW		NO	
S2C0-0150-8	MICROFICHE - 1130 / 1800 NEW CARD	SCRAP**	12861	FEEW		NO	
S2C0-0200-17	MICROFICHE - PRG SYS OTHER THAN OS NEW	SCRAP**	12862	FEEW		NO	
L2C0-0300-2	MICROFICHE - EARLY WARNING PROGRAM PRODUCTS CSP-SCP NEW CARD	SCRAP**	13113	FEEW		NO	
S2C0-0500-9	MICROFICHE - SERVICE AIDS 2025 SA NEW CARD	SCRAP**	13373	FESA		NO	
S2C0-0600-9	MICROFICHE - SERVICE AIDS 2030 SA NEW CARD	SCRAP**	13383	FESA		NO	
S2C0-0666-0	MICROFICHE - ENGINEERING CHANGE ANNOUNCEMENT 2030 ECA FILE CARD		13730	FEECA		NO	
S2C0-0700-9	MICROFICHE - SERVICE AIDS 2040 SA NEW CARD	SCRAP**	13396	FESA		NO	
S2C0-0800-9	MICROFICHE - SERVICE AIDS 2044 SA NEW CARD	SCRAP**	13408	FESA		NO	
S2C0-0900-9	MICROFICHE - SERVICE AIDS 2050 SA NEW CARD	SCRAP**	13416	FESA		NO	
S2C0-0968-0	MICROFICHE - ENGINEERING CHANGE ANNOUNCEMENT 2050 ECA FILE CARD		13731	FEECA		NO	
S2C0-1000-9	MICROFICHE - SERVICE AIDS 2065 SA NEW CARD	SCRAP**	13428	FESA		NO	
S2C0-1100-9	MICROFICHE - SERVICE AIDS 2067 SA NEW CARD	SCRAP**	13441	FESA		NO	
S2C0-1200-9	MICROFICHE - SERVICE AIDS 2075 SA NEW CARD	SCRAP**	13452	FESA		NO	
S2C0-1266-0	MICROFICHE - ENGINEERING CHANGE ANNOUNCEMENT 2075/2085 ECA FILE CARD		13732	FEECA		NO	
G220-2000-0	PROGRAMMING SYSTEMS MEMORANDA (CPSM) 1130 PROGRAMMING SYSTEMS		13602	REFCD	0902	YES	
G220-2001-0	PROGRAMMING SYSTEMS MEMORANDA (PSM) 1800 PROGRAMMING SYSTEMS		13603	REFCD	0901	YES	
G220-2006-7	PROGRAMMING SYSTEMS MEMORANDA (PSM) SYSTEM/360: BPS PROGRAM SYMPTON INDEX	SCRAP**	11966	REFCD	0907	YES	
G220-2007-7	PROGRAMMING SYSTEMS MEMORANDA (PSM) SYSTEM/360: BOS PROGRAM SYMPTON INDEX	SCRAP**	11967	REFCD	0908	YES	
G220-2008-7	PROGRAMMING SYSTEMS MEMORANDA (PSM) SYSTEM/360 : TOS PROGRAM SYMPTON INDEX	SCRAP**	11968	REFCD	0909	YES	
G220-2009-7	PROGRAMMING SYSTEMS MEMORANDA (PSM) SYSTEM/360: DOS PROGRAM SYMPTON INDEX	SCRAP**	11969	REFCD	0910	YES	
G220-2010-7	PROGRAMMING SYSTEMS MEMORANDA (PSM) SYSTEM/360: CONVERSION PROGRAM SYMPTON INDEX	SCRAP**	11970	REFCD	0911	YES	
G220-2011-7	PROGRAMMING SYSTEMS MEMORANDA (PSM) SYSTEM/360: 44PS/BPS PROGRAM SYMPTON INDEX	SCRAP**	11971	REFCD	0912	YES	
S225-3487-1	1259 MAGNETIC CHARACTER READER	SCRAP**	02387	FEMM		NO	
S229-2186-1	PROGRAM IDENTIFICATION LABEL	USE***	13534	PAD		NO	
S229-3185-2	TOOLS AND TEST CE REFERENCE INSERT	SCRAP**	12481	REFCD		NO	
Z320-1046-1	BOIS KWOC INDEX (MARKETING, FIELD ENGINEERING BRANCH OFFICE ADMIN.)	SCRAP**	10821	MANUL	0077	NO	

PRL #13 April 3, 1970

ORDER NUMBER	TITLE/DESCRIPTION	DISPOSITION	ABST#	CLASS	DAPS CODE	SL/ SS	BOL QTY
	MARKETING PUBLICATIONS						
GA24-3545-0	IBM 2956-5 MULTIPOCKET READER SORTER INSTALLATION MANUAL PHYSICAL PLANNING		13844	SRL	0489	NO	ONE
GA27-3014-1	2770 SYSTEM SUMMARY	USE***	13785	SRL	0450	YES	ONE
GC20-1645-1	OUTLINES OF STATISTICAL TECHNIQUES, APPLICATIONS, AND PROGRAMS FOR INDUSTRY, ENGINEERING AND SCIENCE	SCRAP**	02749	DPT		YES	ONE
SC20-1718-0	PROGRAMMED AIRLINES RESERVATION SYSTEM (PARS) LIBRARY DOCUMENTS (VOLUME VII)		13845	TEXT		NO	
SC20-1719-0	PROGRAMMED AIRLINES RESERVATION SYSTEM (PARS) LIBRARY DOCUMENTS (VOLUME VIII)		13848	TEXT		NO	
GC21-7555-0	PROGRAM PRODUCT SPECIFICATIONS IBM SYSTEM/3 CARD RPG II (5701-RG1)		13790	PRGPR		YES	
GC21-7556-0	PROGRAM PRODUCT SPECIFICATIONS IBM SYSTEM/3 CARD UTILITIES (5701-UT1)		13791	PRGPR		YES	
GH20-0136-4	MATHEMATICAL PROGRAMMING SYSTEM/360 (360A-CO-14X) APPLICATION DESCRIPTION	USE***	03692	APPRG		YES	
GH20-0609-1	COURSEWRITER III FOR SYSTEM/360 (360A-UX-01X) VERSION 2 AUTHOR'S GUIDE	USE***	02404	APPRG		YES	
GK20-0338-0	FINANCIAL INFORMATION ON SYSTEM... CITY AND COUNTY OF SAN FRANCISCO		13793	BRIEF		NO	ONE
GK20-0343-0	SYSTEM/360 MECHANIZED ORDER DRESSING AT COLT INDUSTRIES, CRUCIBLE SPECIALTY METALS DIVISION		13796	BRIEF		NO	TWO
GK20-0364-0	THE PRODUCTION INFORMATION AND CONTROL SYSTEM AT J. I. CASE COMPANY, BURLINGTON PLANT PICS, CONTROL CENTER, DATA COLLECTION, PAYROLL		13797	BRIEF		NO	TWO
GN22-0355	TNL FOR C22-6820-10				0489	YES	ONE
GN27-3056	TNL FOR L27-3020-01				0478	YES	ONE
ZR20-8022-2	EDUCATION GUIDE - COURSE CODE S1130	USE***	06516	ED GD		NO	
GV20-4647-0	THE RIGHT ADDRESS		13788	FILM		NO	
GX24-3836-1	0513 0514 ALPHA IBM CONTROL PANEL TEMPLATE	USE***	11228	TEMPL		NO	
GYBO-0512-0	HASP (VERSION 2): LISTINGS		13847	LIST		NO	
GYBO-0515-0	2495 UTILITY FOR OS/360: LISTINGS		13846	LIST		NO	
GY20-0506	TNL Y20-0065-02					YES	
ZZ20-1990-1	DP SALES MANUAL SYSTEMS, MACHINES, RPQ, AND TYPE CATALOG SECTIONS	USE***	12903	MANUL		NO	
ZZ20-1991-1	DP SALES MANUAL TYPE I AND II PROGRAMS AND PROGRAM PRODUCTS SECTIONS	USE***	12901	MANUL		NO	
ZZ20-1992-1	DP SALES MANUAL GI, DATA PROCESSING EDUCATION, SYSTEMS ENGINEERING SERVICES, AND SPECIAL INSTALLATION SERVICES SECTIONS	USE***	12902	MANUL		NO	
ZZ20-1993	TNL Z201990-01 SMR #1				0942	NO	
ZZ20-2078-0	AIR CARGO APPLICATIONS		13792	SSG		NO	TWO
ZZ77-0027-0	OS/360 - HASP IN A MULTIPLE CPU CONFIGURATION USING A SINGLE SHARED WORK QUEUE		12704	PAPER		NO	
ZZ77-0028-0	SYSTEMS ENGINEERING SERVICES: OS/360 MULTIPROGRAMMING PLANNING TASK GUIDE		12712	PAPER		NO	
ZZ77-0029-0	LARGE DATA BASE ARCHITECTURE		12711	PAPER		NO	
ZZ77-0031-0	AN AUTOMATED MUNICIPAL COURT SYSTEM		12709	PAPER		NO	
ZZ77-0034-0	TERMINAL SELECTION CRITERIA IN A TELEPROCESSING ENVIRONMENT		12713	PAPER		NO	
ZZ77-0035-0	ALLOCATION AND SCHEDULING OF SYSTEM ENGINEERING SERVICES		12707	PAPER		NO	
ZZ77-0036-0	MACROS AND CICS ROUTINES - OPERATING SYSTEM (MACROS)		12706	PAPER		NO	
ZZ77-0037-0	RJE - IMPROVEMENT PACKAGE (GERMAN-MODIFICATION)		12705	PAPER		NO	
G320-1066-0	SCHOOLMASTERS TO AN INDUSTRY		13836	BROCH		NO	
G520-2293-0	IBM APPLICATION CUSTOMIZER SERVICE: ACCOUNTS PAYABLE SYSTEM/3 APPLICATION PREPARATION		13795	BROCH		NO	ONE
G520-2294-0	IBM APPLICATION CUSTOMIZER SERVICE: LABOR DISTRIBUTION SYSTEM/3 APPLICATION PREPARATION		13794	BROCH		NO	ONE
G520-2297-0	DOS/360 COBOL SUBSET COMPILER		13838	BROCH		NO	ONE
	FIELD ENGINEERING						
SR25-5501-0	2770 DATA COMMUNICATION SYSTEM		13842	FE ED		NO	
ZR31-0203-3	IBM FIELD ENGINEERING EDUCATION COURSE DEVELOPMENT PROCEDURE INSTRUCTION OUTLINE SUPPORT SHEET	USE***	05156	FORM		NO	
SR31-0478-0	SYSTEM/3 PRESCHOOL		13839	FE ED		NO	
ZR31-0518-0	COURSE ANALYSIS REPORT		13841	FE ED		NO	
SY24-3529-2	2025 PROCESSING UNIT	SCRAP**	04159	FEMDM		NO	
S123-0470-3	MICROFICHE - ILLUSTRATED PARTS CATALOG 2303 DRUM STORAGE	SCRAP**	13647	IPC		NO	
S124-0075-4	MICROFICHE - ILLUSTRATED PARTS CATALOG 1416 INTERCHANGEABLE TRAIN AND CARTRIDGE	SCRAP**	13673	IPC		NO	
S131-0018-0	2970 MODEL 9 BANK TERMINAL		13840	IPC		NO	
S2C0-0250-1	MICROFICHE - EARLY WARNING SYSTEM/3 NEW CARD	USE***	13111	FEEW		NO	
S2C0-1250-9	MICROFICHE - ENGINEERING CHANGE ANNOUNCEMENT 2075 ECA NEW CARD	SCRAP**	13362	FEECA		NO	

* - REPLACED BY NEW ITEM LISTED; SEE OBSOLETE/REPLACE CATALOG.
** - SCRAP PRIOR EDITIONS.
*** - USE PRIOR EDITION.

2914-1 Switching Unit Configuration Chart

The 2914-1 Switching Unit, shown in the RPQ Section of the Sales Manual, provides for many configurations. Most can be determined by using the Chart below. Others can be worked out with Regional Special Product Marketing. The prices must be rechecked with the Sales Manual before quoting.

To use the 2914 Configurator, select the required number of channels and control unit strings along the left side of the chart. Next, read directly across for the total MAC, Purchase Price and MMMC for that configuration. To determine which RPQ's are required and individual prices, use the RPQ number and price directly above the squares containing an "X".

As an example, a 2 X 3 i.e. 2 channels and 8 control unit strings requires RPQ's 880882, 880901, 880904, and 880905. Total price is $746.00 MAC, $36,000 Purchase and $65.00 MMMC.

2914-1 Switching Unit Configuration Chart

CH	CH	RPQ # 880882 Basic MAC $460.	880901 E1 58.	880902 E2 87.	880903 E3 29.	880904 E4 170.	880905 E5 58.	880906 E6 87.	880907 E7 29.	880908 E8 170.	880909 E9 58.	880910 E10 87.	880911 E11 29.	880912 E12 242.	880913 E13 58.	880914 E14 87.	880915 E15 29.	TOTAL MAC	TOTAL Purch.	TOTAL MMMC
2	2	X																$460.	$23,500.	$38.00
2	4	X	X															518.	26,060.	40.50
2	6	X	X			X												688.	34,040.	62.50
2	8	X	X			X	X											746.	36,600.	65.00
4	2	X		X														547.	27,340.	45.50
4	4	X	X	X	X													634.	31,180.	50.50
4	6	X	X	X	X	X		X										891.	43,000.	80.00
4	8	X	X	X	X	X	X	X	X									978.	46,840.	85.00
6	2	X		X						X								717.	35,320.	67.50
6	4	X	X	X	X					X	X							862.	41,720.	75.00
8	2	X		X						X		X						804.	39,160.	75.00
8	4	X	X	X	X					X	X	X	X					978.	46,840.	85.00
2	2																			
2	2	X												X				702.	34,895.	60.00
2	4																			
2	2	X	X											X				760.	37,455.	62.50
2	4																			
2	4	X	X											X	X			818.	40,015.	65.00
4	2																			
2	2	X		X										X				789.	38,735.	67.50
4	2																			
2	4	X		X										X	X			847.	41,295.	70.00
4	2																			
4	2	X		X										X		X		876.	42,575.	75.00
4	4																			
2	2	X	X	X	X									X				876.	42,575.	72.50
4	4																			
2	4	X	X	X	X									X	X			934.	45,135.	75.00
4	4																			
4	2	X	X	X	X									X		X		963.	46,415.	80.00
4	4																			
4	4	X	X	X	X									X	X	X	X	1050.	50,255.	85.00

SPECIAL PRODUCTS INFORMATION

I. Double Word Fetch/MVC

II. Highlights

This feature reduces the LCS access time when fetching sequential addresses and when executing the Move Character Instruction (MVC).

III. Description

The LCS Data Interface is expanded from the present single word (36 bits) to a double word (72 bits). Although the 2050 CPU can process only a single word at a time, each LCS fetch cycle buffers a double word and its address. If the address of the next LCS fetch request is equal to the address of the buffered double word, the data is cycled from the buffer and a LCS cycle is not taken. The data in the buffer becomes invalid if a store operation or test and set instruction is executed and the address is equal to the address of the buffered word. To eliminate a possible programming problem when in a shared storage configuration, the buffer is reloaded from LCS after two consecutive fetches from the same double word address. The execution time of the Move Character (MVC) Instruction is reduced when the address of the first or second operand or both is a LCS address. The even word of a double word to be stored is buffered until the odd word is available and the double word is stored in LCS during one cycle. The double word fetch buffer is made use of for any fetches during the execution of a MVC Instruction.

IV. Control Switch

A four position switch is provided to disable the feature to allow the double word buffer and its address to be displayed. When the feature is disabled a LCS cycle will be taken for each word stored or fetched. With the switch in the display position, either word of the double word buffer or its address can be displayed.

V. Storage Size

All standard Model 50 high speed storage sizes are available with the 2361 Model 1 or 2 LCS units. A maximum of two 2361's may be attached with this feature.

VI. Prerequisite

FC 8080 on 2050
RPQ 888157 on 2361 and one of the following specified features: 9001, 9006, 9016, 9021, or 9036.

VII. Reference RPQ* Numbers and Area Prices

		MAC	Purch.	MMMC
E68304	Double Word Fetch/MVC	$776.	$27,936.	$8.50

*All RPQ prices shown are "area" prices. Firm quotations must be obtained from Regional Special Product Marketing before quoting to the customer.

April 24, 1970 **IBM INSTALLATION NEWSLETTER** Issue No. 70-08

Table of Contents

OS/360 LINKAGE EDITOR ENQUEUE ON OUTPUT DATA SETS

In Release 18, the Linkage Editor gained the ability of functioning in more than one region or partition of a multiprogramming environment, simultaneously using the same SYSLMOD output data set without conflict. This enhancement was implemented, in response to an APAR in the normal course of maintenance, using the ENQUEUE/DEQUEUE function.

OS/360 MFT ABEND OUTPUT IMPROVEMENT

This contribution has not been submitted to any formal IBM test. Potential users should evaluate its usefulness in their own environment prior to implementation.

Shown below is a Superzap change to module IGC0005A to increase the number of printed lines per page from 56 to 75. This is particularly useful in installations which use their 1403 set at 8 lines per inch. This change is for OS Release 18 and may be release dependent.

```
  NAME IGC0005A
  DUMP IGC0005A  IGC0005A
  VERIFY 02DA 41100038
  REP 02DA 4110004B
  DUMP IGC0005A  IGC0005A
//SYSLIB DD DSN=SYS1.SVCLIB,DISP=OLD
```

OS/360 VOLUME CYCLING WITH GENERATION DATA GROUPS

This contribution has not been submitted to any formal IBM test. Potential users should evaluate its usefulness in their own environment prior to implementation.

Many installations require that a specific series of pre-determined numbered tapes be used in a cyclical rotation as the residency for a particular master data set. Under OS/360, the cataloging facility affords the user the option of allowing the system to keep track of the physical residency and volume serial numbers of permanent data sets.

However, cataloging does not furnish a capability to rotate specific tapes as data set residencies in a son-father-grandfather-great grandfather relationship, where tapes A, B, C and D become D A B and C, C D A and B, B C D and A and eventually A B C and D. Operators must pre-determine and physically ensure that the oldest tape in the cycle is mounted when a system request is made to create a new master data set.

Through the use of Job Control Language and Generation Data Groups, the facility of rotating specific tapes may be easily and automatically achieved. The example illustrates a data set, the most current of which is on tape A and the oldest of which is on tape D, whose newest updated version should be automatically created on tape D, while tapes A, B and C are "pushed down".

Status of catalog before data set update:

```
 0  Tape A - Son
-1  Tape B - Father          } DSN=SYS1.TAPE
-2  Tape C - Grandfather     }
-3  Tape D - Great Grandfather
```

JCL within job step creating new data set:

```
//MAGIC DD DSN=SYS1.TAPE(-3),DISP=OLD
//OLDMAST DD DSN=SYS1.TAPE(0),DISP=OLD
//NEWMAST DD DSN=SYS1.TAPE(+1),
  DISP=(NEW,CATLG),
// VOL=REF=*.MAGIC,DCB=..., etc.
```

where the referral for the tape onto which the new version of the data set is to be written will always cause the automatic mounting of the oldest tape.

Status of catalog after data set update:

```
 0  Tape D - Son
-1  Tape A - Father
-2  Tape B - Grandfather
-3  Tape C - Great Grandfather
```

1130 FORTRAN IMPROVED WITH COMMERCIAL SUBROUTINE

This technique may be of interest to engineering oriented 1130 shops where the Commercial Subroutine Package ordinarily is not used. It has not been submitted to any formal IBM test. Potential users should evaluate its usefulness in their own environment prior to implementation.

Incorporation of the Commercial Subroutine CALL READ to read from the 1442 in place of the FORTRAN READ has improved the run time of an engineering job from six hours to one hour and ten minutes on an 8K 1130. The FORTRAN READ was limiting total job throughput to the card speed of 36 cpm. With the Commercial Subroutine, the card read speed attained 200 cpm. FORTRAN statements were used to produce output by placing output on intermediate disk storage, then linking to a subroutine at the end of the job and using FORTRAN WRITES to print the output.

All Type IV and Prior Use programs which appear in this Installation Newsletter were accepted prior to December 31, 1969.

TYPE III AND IV PROGRAMS

Type III programs are those which have been submitted by one or more IBM employees. They are programs of general interest submitted for unrestricted distribution. They have met a basic set of programming and documentation standards but are not program tested in any formal fashion by the IBM Corporation. The user is expected to make the final evaluation as to their usefulness in his own environment. These programs were available for delivery from the Program Information Department prior to June 23, 1969.

Type IV programs are those contributed for unrestricted distribution by one or more authors of which at least one is an employee of an IBM customer. They are made available by IBM essentially in the author's original form, but conform to published Type IV standards. IBM exercises no control over the technical content of the documentation but merely assures that the quality of reproduction is satisfactory. Type IV programs have not been tested by IBM. The user is expected to make the final evaluation as to their usefulness in his own environment.

IBM MAKES NO WARRANTY, EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE AS TO THE DOCUMENTATION, FUNCTION OR PERFORMANCE OF THESE PROGRAMS.

TYPE III PROGRAMS WITH SERVICE A CLASSIFICATION

Type III programs which were given Service A Classification, perform functions which may be fundamental to the operation and maintenance of the user's system. These programs have not been subjected to formal test by IBM.

Until reclassified, IBM will provide for these Type III programs with the following: (a) Central Programming Service including design error correction and automatic distribution of corrections; (b) Field Engineering Programming Service including design error verification, Authorized Programming Analysis Report (APAR) application of Program Temporary Fixes or development of an emergency by-pass when required.

IBM does not guarantee service results or represent or warrant that all errors will be corrected. The user is expected to make the final evaluation as to the usefulness of these programs in his own environment.

THE FOREGOING IS IN LIEU OF ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

PRIOR USE PROGRAMS

There are a limited number of programs that were developed prior to June 23, 1969 which were in the possession of and were being used by customers without restriction, but were not included in the Type III or IV libraries. These programs have been designated Prior Use Programs and are being made available through the Program Information Department (PID).

Prior Use Programs have not been developed or tested in any formal fashion by the IBM Corporation, and, therefore, are available only on an "as is" basis without charge. It is the customer's responsibility to make the final evaluation as to the usefulness of the program in his own data processing and business environment.

IBM MAKES NO WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE AS TO THE DOCUMENTATION, FUNCTION OR PERFORMANCE OF THESE PROGRAMS.

Prior use programs are classified as follows:

360K - System/360 Model 20
360L - System/360 Models 25 and Above
1131 - 1130 Computing System
1801 - 1800 Data Acquisition and Control System
1402 - 1401 Data Processing System

---------- TYPE III REVISION ----------

1401-10.3.063 FAST-FULLY AUTOMATIC SCORING TECHNIQUE. Material Revised: The Documentation and Machine Readable.

---------- NEW TYPE IV PROGRAM ----------

AN 1130 HIGH SPEED BISYNCHRONOUS COMMUNICATIONS SYSTEM

DESCRIPTION - The High Speed Binary Synchronous Communications subroutine is an 1130 program which provides high-level capabilities for tele-communications with a S/360 or another 1130 over a point-to-point high speed (40.8K BAUD) half duplex line.

The system provides support at the GET/PUT level for an arbitrary number of users (message destinations) in either machine. Messages containing data in any format can be sent from any user in one machine to any user in the other, with the routing automatically controlled by the system.

PROGRAMMING SYSTEMS - Written in 1130 DMPS. The package is designed to run in conjunction with the Type 4 S/360 program entitled "A High Speed Bisynchronous Communications Access Method" (360D-06.3.012).

MINIMUM SYSTEM REQUIREMENTS - The 1130 configuration requires RPQs 831551 (expansion 1133) and F21059 (communications adapter).

BASIC PROGRAM PACKAGE
DOCUMENTATION - Write-up.
MACHINE READABLE - Source code.

OPTIONAL PROGRAM PACKAGE - None.

ORDERING INFORMATION: PROGRAM NUMBER 1130063005

	PROGRAM NUMBER EXTENSION	DISTRIBUTION MEDIUM TYPE	CODE	USER VOLUME REQUIREMENT
BASIC	none	DTR*	00	none
OPTIONAL	none	none		none

ACCUMULATIVE LIST OF TYPE III AND IV PROGRAMS

Following is an accumulative list of Type III and Type IV programs which have been made available since the last edition of the Catalogs of Programs or their Supplements. Abstracts describing these programs can be found in the Installation Newsletter Issue No. shown below. All programs which appear in the accumulative list were accepted prior to December 31, 1969.

Program Number	Program Title	Newsletter Issue No.
360D-00.0.023	LISTUTIL	70-03
360D-01.4.011	ZEROX - COMMUNICATIONS PRODUCTS DIVISION OS/360 MFT ACCOUTING ROUTINES	70-01
360D-02.0.004	INDEX SEQUENTIAL ERROR BYTE CHECK ROUTINE	69-24
360D-03.0.014	MULTIPROGRAMMING SYSTEM (MPS)	70-05
360D-03.3.012	SPEAKEASY	70-04
360D-03.4.033	A 2250 MODEL 1 SIMULATION SUPPORT PACKAGE	70-02
360D-03.4.034	PS 44/2314 SUPPORT	70-04
360D-03.5.005	A SYSTEM TO PROCESS ABSTRACT CATALOGS AND RELATED INDICES	70-02
360D-03.5.006	RDC/1 REPORT DATA ON CONDITION	70-03
360D-03.5.007	REPORT WRITER	70-06
360D-03.7.033	BINARY SEARCH MACRO	70-04
360D-05.0.003	OS MASTER SCHEDULER INITIALIZATION IMPROVEMENTS	69-25
360D-05.1.019	O.S./M.I.S.P. VERSION 1.1	70-06
360D-05.1.020	PITT TIME SHARING SYSTEM AND ASSOCIATED LANGUAGE PROCESSORS	70-06
360D-05.2.012	44 MFT MULTIPROGRAMMING SYSTEMS FOR THE 360 MODEL 44	70-02
360D-05.2.013	DOS/RESQUEUE - RESOURCE QUEUING AND USER SVC'S	70-05
360D-06.3.013	1907 REMOTE ORDER ENTRY PROGRAM	70-02
360D-06.3.014	PHARMACY PROCESSING IN A HOSPITAL INFORMATION SYSTEM WITH MISP	70-04
360D-06.3.015	NURSING INSTRUCTION ORDER PROCESSING IN A MEDICAL INFORMATION SYSTEMS ENVIRONMENT	70-03
360D-06.7.025	MISP LABORATORY REQUISITION SYSTEM	70-03
360D-06.8.002	LPI	69-24
360D-08.6.011	PNRG, PERSPECTIVE PLOTTING ROUTINE, ARBITRARY GRID	70-06
360D-08.6.012	PRG, PERSPECTIVE PLOTTING ROUTINE, RECTANGULAR GRID	70-06
360D-08.6.013	PLT360, IBM 1627 PLOTTING ROUTINE	70-06
360D-11.4.003	THE 141 MONITOR - AN EDUCATIONAL COMPUTER FOR BASIC COMPUTER INSTRUCTION	70-03
360D-12.1.023	CEFF - S/360 PIOCS PROGRAM TO COPY EMULATOR FOURTEEN HUNDRED FILES FROM 2311'S TO 2314'S	70-06
360D-15.2.013	SYSTEM/360 COMPUTER ASSISTED MENU PLANNING	70-03
360D-15.3.002	QPS: A QUADRATIC PROGRAMMING SYSTEM FOR THE IBM/360.	70-06
360D-15.5.003	QUAKER OATS VERSION OF PURDUE SUPERMARKET MANAGEMENT GAME	70-06
360D-16.2.027	ICES/360 SOURCE STRUDL-II	70-02
360D-16.2.028	ICES/360 STRUDL - II - CSAS16	70-04
360D-16.3.002	PULSE TESTING VIA THE FAST FOURIER TRANSFORM	70-01
360D-16.5.002	GENERAL THERMAL PIPE STRESS AND DEFLECTION PROGRAM	70-03
360D-19.7.012	A 2980-IV ON-LINE FINANCIAL SYSTEM	70-02
360D-23.4.003	GRAPHIC PART PROGRAMMER	70-04
360D-40.0.002	INTEGER MULTIPLE PRECISION SUBROUTINE PACKAGE FOR THE IBM 360	70-03
360D-42.0.002	MINFUN FUNCTION MINIMIZER	70-03
360D-99.0.008	LIBRARY CIRCULATION CONTROL SYSTEM	70-05
360E-03.6.003	IBM S/360 EXPANDED CARD EDITING PROGRAM	70-03
360E-12.2.004	UNIVAC TO IBM RPG TRANSLATOR	70-02
1130-00.0.011	CDUTL, A GENERAL PURPOSE CARD UTILITY	69-25
1130-00.0.013	80/80 LIST PROGRAM	70-04
1130-01.0.001	DCODE - AN OBJECT - TO - SOURCE LISTING PROGRAM FOR THE IBM 1130	70-02
1130-01.0.002	INTERRUPT REQUEST INTERCEPTION SUBROUTINE	70-04
1130-03.1.001	SIMULATED ASSEMBLY PROGRAM	70-02
1130-03.4.014	A 2250 MODEL 1 SIMULATION SUPPORT PACKAGE	70-02

1130-03.4.015	EXECUTION-TIME FORMAT STATEMENTS	70-02
1130-03.4.016	FORTRAN I/O ERROR TRAP - SFIO AND SFIOP	70-06
1130-04.0.002	ASSEMBLER LANGUAGE DEBUGGING PROGRAM SEGMENTS	70-04
1130-04.2.004	ASSEMBLER/FORTRAN PROGRAM TRACE PACKAGE	70-04
1130-05.3.002	IPL CARD READING PROGRAM	70-04
1130-06.3.005	AN 1130 HIGH SPEED BISYNCHRONOUS COMMUNICATIONS SYSTEM	70-07
1130-06.6.018	TYPE SUBROUTINE FOR LOWER-CASE OUTPUT ON 1130 CONSOLE PRINTER	70-02
1130-07.0.003	DATA SWITCH INPUT SUBROUTINE	70-04
1130-08.0.003	PRNTZ - 1132 PRINTER OUTPUT SUBROUTINE	70-02
1130-08.7.001	PRINTER GRAPHIC SUBROUTINE PACKAGE	70-04
1130-15.4.008	1130-8K STUDENT SCHEDULING SYSTEM	70-02
1130-17.3.004	FAST	69-25
1130-23.4.006	RCAPS - AN 1130 CONTOURING NUMERICAL CONTROL PROCESSOR	70-04
1130-40.0.001	INTEGER MULTIPLE PRECISION SUBROUTINE PACKAGE FOR THE IBM 1130	70-02
1130-45.1.001	MATRIX INVERSION SUBROUTINE	70-04
1130-99.0.002	BILLERICA 1130 SCHEDULER	70-05
1130-99.0.003	SCHOOL ATTENDANCE PACKAGE FOR THE IBM 1130 USING DISK STORED STUDENT RECORDS	70-05
1500-01.6.001	RAS/1130 COURSEWRITER II REASSEMBLE PROGRAM	69-24
1500-03.8.001	RESPONSE ANALYSIS FUNCTIONS FOR THE 1500 SYSTEM	70-03
1500-99.0.002	FUNCTION EP 1500 CW II FUNCTION CAI SYSTEM	69-24
1500-99.0.003	U.T. COURSEWRITER FUNCTION PACKAGE	70-03
1500-99.0.004	FUNCTION KMIN 1500 CWII FUNCTION CAI SYSTEM	70-05
1800-01.8.001	BINARY DATA ARRAY MULTIPLEXER ROUTINE	70-06
1800-03.4.009	REWIND AND UNLOAD SUBROUTINE	70-06
1800-06.7.001	LOGICAL AND RELATIONAL PACKAGE	70-06
1800-07.2.001	VARIABLE FORMAT SUBROUTINE	70-06
1800-43.2.001	CONTROL SYSTEMS ANALYSIS PROGRAM	70-04
1401-10.3.071	1401 UNIVERSITY ADMISSIONS INFORMATION SYSTEM	69-14
1401-10.3.072	FINANCIAL AID INFORMATION SYSTEM (LOAN SUBSYSTEM)	69-14
1401-10.3.073	FINANCIAL AID INFORMATION SYSTEM (STUDENT AID SUBSYSTEM)	69-14
1620-01.6.160	LABELR - A REPORT GENERATOR	69-15
1620-02.0.066	A MODIFIED PDQ FORTRAN SYSTEM	69-16
7040-H2 3552	MULTIDIMENSIONAL OPTIMIZATION USING PATTERN SEARCH	69-11
7094-F2 3554	EIGENVALUE - EIGENVECTOR ROUTINE FOR REAL COMPLEX MATRIX	69-15

ACCUMULATIVE LIST OF PRIOR USE PROGRAMS

Following is an accumulative list of prior use programs which have been made available since the last edition of the Catalogs of Programs or their Supplements. Abstracts describing these programs can be found in the Installation Newsletter Issue No. shown below.

Program Number	Program Title	Newsletter Issue No.
360L-03.2.001	BUFF 40	70-04
360L-04.4.001	1827 CONTROL PROGRAM	70-04
360L-05.0.001	2260 SUPPORT FOR MISP	70-05
360L-06.7.001	INFORMATION RETRIEVAL MANAGEMENT SYSTEM (IRMS)	70-06
360L-05.1.001	ONLINE/OS - A CONVERSATION MONITOR FOR OS/360	70-04
360L-13.0.001	SASE IV - AN IMPROVED PROGRAM FOR THE STATISTICAL ANALYSIS OF SERIES OF EVENTS	70-05
360L-15.0.001	DZLP ADAPTIVE BINARY PROGRAMMING	70-04
360L-16.3.001	IBM 360 DYE SELECTION AND FORMULATION PROGRAM FOR COLOR MATCHING	70-06
360L-19.0.001	BROKERAGE COMMUNICATIONS CONTROL APPLICATION PROGRAM	70-04
360L-19.0.002	IBM S/360 LOS ANGELES FINANCIAL ON-LINE-BANKING REFERENCE SYSTEM FOR SAVINGS	70-04
360L-23.1.001	PAPER MACHINE TRIM SYSTEM	70-06
360L-28.3.001	CMS - CREDIT MANAGEMENT SYSTEM	70-07
360L-30.0.001	LEGISLATIVE INFORMATION SYSTEM DEMONSTRATION PACKAGE (LEGISY)	70-04
360L-50.1.001	ALIS STARTER SYSTEM	70-04
360L-99.0.001	AUTOMATIC DRIVER RE-TEST PROGRAM S/360	70-05
360K-19.0.001	MODEL 20 BROKERAGE PACKAGE	70-04
1131-06.6.001	LINOFILM SUPER QUICK OUTPUT MODULE FOR 1130 PHOTO COMPOSITION PROGRAM	70-04
1131-16.2.001	TRAFFIC CONTROL PROGRESSION PROGRAM	70-05
1131-21.1.002	1130 4k PAYROLL PACKAGE	70-04
1801-03.4.001	MODIFICATIONS TO INCLUDE THE 1132 PRINTER IN THE IBM 1800 MULTIPROGRAMMING SYSTEM (MPX) 1800-OS-010	70-04
1801-05.1.001	1800 PALO ALTO LABORATORY SYSTEM (PALS) VERSION II.	70-04
1801-23.1.001	PAPER MACHINE TRIM SYSTEM	70-05
1402-10.3.075	SOCRATES STUDENT SCHEDULING - BETA VERSION	70-04

IBM Installation Newsletter

June 19, 1970 Issue No. 70-12

CONTENTS

(G means Green Section)

Distribution: By DAPS Subscription.
Order Number 0925.

*Requires Immediate Attention

ZZ20-7012 Printed in U.S.A.

SELECTED SHORT SUBJECTS

The purpose of Selected Short Subjects is to bring together and highlight concise, factual and timely information indicating that action is to be taken by the IBM representative whose accounts are affected.

1. NOTICE TO READERS AND CONTRIBUTORS

I'm sure there has been some uncertainty in the field about the publishing of code in the INL. Now the situation has been clarified, and the door has been opened. The INL has been identified as one of the vehicles to be used in order to assure the free interchange of data processing ideas, concepts, know-how and techniques, and to avoid duplications of effort.

I'm referring to Marketing Announcement 270-39, which provides guidelines for the use, interchange and submission of programming. Each of you should read this carefully. It is good news!

Contributions lately have been rather slow. Now I expect more activity. All readers of the INL are invited to become contributors!

Editor

2. NEW RELEASES OF PROGRAMS

This information is furnished in order to assure that you are aware of new releases of programs which are being used by your customer. The programming service period for program releases being replaced by new releases is stated below.

1. Version 1 Modification Level 1 of the IBM Customer Information Control System (5736-U11) is available from PID. Version 1 Modification Level 0 will remain "current" for programming service purposes until August 14, 1970. Programming service classification: B.

2. Version 2 Modification Level 1 of the Input/Output Control System for the Binary Synchronous Communications Adapter (360T-CQ-111) is now available from PID. Version 2 Modification Level 0 will remain "current" for programming service purposes until August 12, 1970. Programming service classification: A.

3. Release 9 of the IBM System/360 Model 20 Disk Programming System (DPS) is available from PID. Version 5 Modification Level 0 will remain "current" for programming service purposes until August 4, 1970. Programming service classification: A.

4. Version 2 Modification Level 4 of System/360 AUTOSPOT Numerical Control Processor (360A-CN-08X) is available from PID. Version 2 Modification Level 3 will remain "current" for programming service purposes until August 29, 1970. Programming service classification: B.

5. Version 2 Modification Level 1 of Shared Hospital Accounting System (360A-UH-11X) is available from PID. Version 2 Modification Level 0 will remain "current" for programming service purposes until August 18, 1970. Programming service classification: B.

6. System Release 22/23 of DOS is available from PID. System Release 21 will remain "current" for programming service purposes until August 15, 1970. System Release 21 publications will remain available as long as Release 21 remains "current". These publications are available from Mechanicsburg only. Programming service classification: A, except for AUTOTEST, which has programming service classification C.

7. Version 1 Modification Level 1 of APL/360-OS (5734-XM1) is available from PID. Version 1 Modification Level 0 will remain "current" for programming service purposes until August 3, 1970. Programming service classification: Modified B.

CALL/360-OS PTFs, UTILITIES AND EDUCATION

1. Two PTFs are available for CALL/360-OS. These should be applied before the system is put into production. Contact FE.

2. An additional utility function is available with the initial release of CALL/360-OS that is not clearly defined in the current documentation. This utility allows FORTRAN, BASIC or PL/I programs to be introduced into the CALL/360 system via the background. Existing programs can be loaded into CALL/360-OS libraries off-line and debugged from a terminal. This utility (U#UPDATE) will aid in converting existing programs to the on-line mode of operation. Details are being sent to your SD&I Center. A TNL will reflect this information.

3. A customer class on Installing CALL/360-OS (S-3700) is planned for August 4 - 7, 1970, in Chicago. See SECOM for details.

4. "It's CALL/360-OS, whenever discriminating problem solvers congregate."

DOS/360 PRIVATE LIBRARY CREATION DURING SYSGEN

This contribution has not been submitted to any formal IBM test. Potential users should evaluate its usefulness in their own environment prior to implementation.

The DOS PID distribution is made on three volumes for Release 23. Vol. 1 is a Core Image and Relocatable Library system, Vol. 2 is a Core Image and Source Statement Library system, and Vol. 3 is a Private Source and Private Relocatable Library system. It may be advantageous for your to redefine Vol. 2 as a Private Source Statement Library. The following technique may be used to create a private library from an existing SYSRES volume.

A. Display the System Directory to find the library beginning and ending cyclinders.

```
// JOB STEP1
// EXEC MAINT
/&
```

B. Create a label in the VTOC to define the Source Statement Library extent. (Reply DELETE to the 4444 MSG.)

1. Assemble the following job.

```
       PUNCH    '     Phase LABCRT,S'
LABCRT START 0
DASD   DTFSD   BLKSIZE=100,IOAREA1=DASDIO,
                        TYPEFLE=OUTPUT
BEGIN  BALR          2,0
       USING         *,2
       OPEN          DASD
       EOJ
DASDIO DS            CL100
       END           BEGIN
```

2. Link and execute the program from item 1 above.

```
// JOB CREATE A VTOC ENTRY
// ASSGN SYS007,X'192'
// DLBL DASD, 'DOS.PVT.LIBRARY.NO.2',99/365
// EXTENT SYS007,111111,1,0,NNN,NNN
// OPTION LINK
   **OBJECT DECK FROM ITEM 1**
// EXEC LINKEDT
// EXEC
/&
```

C. Assign the new volume as a Private Library.

DOS/360 POWER II WITH NON-ENDING PROGRAMS

This contribution has not been submitted to any formal IBM test. Potential users should evaluate its usefulness in their own environment prior to implementation.

This modification, illustrated in Exhibit 1, allows POWER II users to run non-ending programs in the POWER partition. By using simple operator commands, he may request that a program be loaded and share the partition with POWER and at a later time request removal of the program. When either POWER or the user program go into a WAIT, the other is given control when both programs are in a WAIT, the partition is placed in a WAIT.

A DOS Supervisor with multitasking capability (AP=YES) is not required; the user program does not run as a DOS subtask. Therefore, the user program must be non-ending. If it terminated, POWER would be terminated also. A program may effect removal from the system by issuing a termination request message to the console and then waiting on a dummy CCB. The operator then issues the REMOVE command.

An additional parameter (SUBTASK) and operator command (X) are added to POWER.

PARAMETER

SUBTASK = {**NO** / **(YES,phase 1,phase 2,...phase n)**,}

NO: Non-ending programs and the X operator command not supported.

YES: Non-ending programs (phase 1, through phase n) and the X operator command are supported.

Phase 1 through phase n: Eight character phase names of all non-ending programs cataloged to run in the partition with POWER. The X operator command processor verifies that any phasename entered appears in this list.

X OPERATOR COMMAND

X	A	Display active phasename (if any)
X	LOAD,phasename	Load and activate non-ending program 'phasename'
X	REMOVE,phasename	Deactivate non-ending program 'phasename'

phasename: Eight character phasename of non-ending program to be run in partition with POWER. Must appear in the SUBTASK parameter when POWER is assembled. Number of entries is limited only by assembler Operand Sublist Length limitation.

```
// JOB UPDATE POWER II                                          ) $ ) .
// EXEC MAINT                                                   ) $ ) .
 UPDATE A.POWER                                                 )A$ )I.
) REP 9                                                         )A$ )I.
               &PRIORTY=NO,                                     )A$ )I.X
               &SUBTASK=NO,  SUBTASK ADDED --    03/30/70       )A$ )I.X
) ADD 383                                                       )A$ )I.
         AIF   ('&SUBTASK' EQ 'NO').N17                         )A$ )I.
         AIF   ('&SUBTASK' NE 'YES').XPROC1                     )A$ )I.
         AIF   ('&SUBTASK(2)' NE '').N17                        )A$ )I.
.XPROC1  ANOP                                                   )A$ )I.
         MNOTE ''''SUBTASK'' ENTRY INVALID'                     )A$ )I.
&M99     SETB  (1)                                              )A$ )I.
) ADD 459                                                       )A$ )I.
         PUNCH ' PHASE FGPTYPXX,+0 '                            )A$ )I.
         PUNCH ' INCLUDE ,(FGPTYPXX) '                          )A$ )I.
) REP 593                                                       )A$ )I.
         POWERSUP READER=&READER,                               )A$ )I.X
               SUBTASK=&SUBTASK  SUBTASK ADDED --  03/30/70     )A$ )I.
) REP 595                                                       )A$ )I.
         POWERCOM READER=&READER,PRIORTY=&PRIORTY,              )A$ )I.X
               SUBTASK=&SUBTASK SUBTASK ADDED--  03/30/70       )A$ )I.
) END                                                           )A$ )I.
```

EXHIBIT 1

```
 UPDATE A.POWERSUP                                                    )A$ )I.
) REP 1                                                               )A$ )I.
         POWERSUP &READER=TYPE1,                                      )A$ )I.X
               &SUBTASK=NO   SUBTASK ADDED --  03/30/70               )A$ )I.
) REP 156                                                             )A$ )I.
         AIF   ('&SUBTASK' EQ 'NO').RWF03                             )A$ )I.
*************************************************************       )A$ )I.
*                                                                     )A$ )I.
*     THE FOLLOWING MULTI-TASKING CODE HAS BEEN ADDED                 )A$ )I.
*                              03/30/70                               )A$ )I.
*                                                                     )A$ )I.
*************************************************************       )A$ )I.
SVCFG    CLI   35,7                        WAIT ISSUED                )A$ )I. ****
SVCFGX   B     SVCFG1 CHANGED TO -BNE- WHEN 1ST SUBTASK ACTIVATE      )A$ )I.
         TM    33,X'F0' WAS IT A WAIT FROM SUPVR                      )A$ )I.
         BZ    EXITA YES, GO LET SUPVR HANDLE                         )A$ )I.
         CLI   SVC7SW,X'FF'              WHERE DID SVC7 COME FROM     )A$ )I. ****
         BE    SVC7RET   FROM SVC7 RETURN FROM SUPVR                  )A$ )I.
         TM    2(1),X'80'                WAIT ALREADY SATISFIED       )A$ )I. ****
         BO    EXITA                     YES. RETURN                  )A$ )I. ****
         CLI   SUBTSKSW,X'FF'            WHICH TASK ISSUED SVC7       )A$ )I. ****
         BE    SUBTSK                    SUB-TASK                     )A$ )I. ****
         STM   10,7,SVMREGS+8      SAVE POWER REGS 10-7               )A$ )I.
         MVC   SVMREGS(8),0              SAVE 8-9                     )A$ )I. ****
         MVC   SVMPSW(8),32              SAVE PSW                     )A$ )I. ***'
TSTSTSK  LM    10,7,SVSREGS+8  RESTORE SUBTASK REGISTERS              )A$ )I.
         MVC   0(8,0),SVSREGS            REGS 8,9                     )A$ )I. ****
         MVI   SUBTSKSW,X'FF'            SET SUB-TASK INDICATOR       )A$ )I. ****
TMINSTR  TM    2(1),X'80' SUB-TASK COMPLETED I/O                      )A$ )I.
         BZ SETSVC7 BRANCH IF SUB-TASK I/O NOT COMPLETE               )A$ )I.
TSKSELS  MVC   8(8,0),SVSPSW             RESET PSW                    )A$ )I. ****
         B     EXITAB                    RETURN TO SUB-TASK           )A$ )I. ****
*                                                                     )A$ )I. ****
SETSVC7  MVI   SVC7SW,X'FF'          SET MULTI-TASK SVC7 BOUND SW     )A$ )I.
         MVC   37(3,0),=AL3(SVC7+10) FORCE SVC7 RETURN TO SVC7        )A$ )I.
*           MAKE INSTR COUNTER IN SUPVR CALL OLD PSW PT TO TEN        )A$ )I.
*           BYTES AFTER THE SVC7.  DOS WILL BACK THIS UP 10           )A$ )I.
*           BYTES AND STORE IT AS THE IC IN THE POWER PARTITION       )A$ )I.
*           SAVE AREA PSW.                                            )A$ )I.
         B     EXITA               GO TO SUPVR SVCHDLR               )A$ )I.
*           THE POWER PARTITION WILL NOW BE MADE SVC7 BOUND.          )A$ )I.
*           WHEN AN I/O INTERRUPT FOR THE POWER PARTITION             )A$ )I.
*           OCCURS, THE SVC7 WILL BE EXECUTED AND SVCFG               )A$ )I.
*           HANDLE IT.                                                )A$ )I.
*                                                                     )A$ )I. ****
SUBTSK   STM   10,7,SVSREGS+8            SAVE SUB-TASK REGS           )A$ )I. ****
         MVC   SVSREGS(8),0              REGS 8,9                     )A$ )I. ****
         MVC   SVSPSW(8),32              SAVE PSW                     )A$ )I. ****
TSTMTSK  LM    10,7,SVMREGS+8            RESTORE MAIN REGS            )A$ )I. ****
         MVC   0(8,0),SVMREGS            REGS 8,9                     )A$ )I. ****
         MVI   SUBTSKSW,X'00'            SET MAIN SWITCH              )A$ )I. ****
         TM    2(1),X'80'                I/O COMPLETED                )A$ )I. ****
         BZ    SETSVC7                   NO                           )A$ )I. ****
TSKSELM  MVC   8(8,0),SVMPSW             RESTORE MAIN PSW             )A$ )I. ****
         B     EXITAB                    RETURN TO MAIN TASK          )A$ )I. ****
*                                                                     )A$ )I. ****
SVC7RET  MVI   SVC7SW,X'00'   CLEAR MY SVC7 BOUND SWITCH              )A$ )I.
         TM    2(1),X'80'              WAIT SATIFIED FOR THIS TASK    )A$ )I.
         BO    TSKSEL                    YES. GO FIND WHICH TASK      )A$ )I. ****
         CLI   SUBTSKSW,X'FF'            SUB-TASK                     )A$ )I. ****
         BE    TSTMTSK                   YES- GO TEST MAIN            )A$ )I. ****
         B     TSTSTSK                   GO TEST SUB-TASK             )A$ )I. ****
```

EXHIBIT 1 cont'd.

For IBM Internal Use Only

```
TSKSEL   CLI   SUBTSKSW,X'FF'          SUB-TASK                    )A$ )I. ****
         BE    TSKSELS              YES- GO TASK SELECT SUB-TASK )A$ )I.
         B     TSKSELM                 GO TASK SELECT MAIN      )A$ )I. ****
*                                                                 )A$ )I. ****
         LTORG                                                    )A$ )I. ****
SVC7     SVC   7                  RETURN FROM SUPVR(BOTH TASKS)  )A$ )I.
SVC7SW   DC    X'00'  FF WHEN BOTH TASKS SVC7 BOUND               )A$ )I.
SUBTSKSW DC    X'00' FF SUBTASK RUNNING -- 00 POWER RUNNING      )A$ )I.
SVMPSW   DS    2F                  POWER PSW                    )A$ )I. ****
SVMREGS  DS    16F                 POWER REGS                   )A$ )I. ****
SVSPSW   DS    2F                  SUB-TASK PSW                 )A$ )I. ***'
SVSREGS  DS    16F                 SUB-TASK REGS                )A$ )I. ****
XPROPHAS DC    8X'40'              ACTIVE PHASE NAME            )A$ )I.
FIRSTIME MVC   TMINSTR(4),HLDINSTR REPLACE B INSTR WITH TM      )A$ )I.
         LA    1,XPROPHAS PT TO PHASE NAME                      )A$ )I.
         SR    0,0 .                                            )A$ )I.
         MVI   35,1 CHANGE SVC CODE TO FETCH                    )A$ )I.
         B     EXITA .                                          )A$ )I.
HLDINSTR TM    2(1),X'80' SUBTASK COMPLETED I/O                 )A$ )I.
XPROINST B     FIRSTIME .                                       )A$ )I.
*                                                               )A$ )I. ****
         LTORG                                                  )A$ )I. ****
* END OF ADDED CODE*************************************************)A$ )I.
SVCFG1   CLI   35,WAKEUP                 SPECIAL SVC REQUEST     )A$ )I.
         AGO   .RWF04                                           )A$ )I.
.RWF03   ANOP                                                   )A$ )I.
SVCFG    CLI   35,WAKEUP             SPECIAL SVC REQUEST        )A$ )I.
.RWF04   ANOP                                                   )A$ )I.
) END                                                           )A$ )I.
 UPDATE A.POWERRES                                              )A$ )I.
) ADD 628                                                       )A$ )I.
         DC    C'X'                                             )A$ )I.
) END                                                           )A$ )I.
 UPDATE A.POWERCOM                                              )A$ )I.
) REP 1                                                         )A$ )I.
         POWERCOM &READER=TYPE1,&PRIORTY=NO,                    )A$ )I.X
               &SUBTASK=NO SUBTASK ADDED--            3/30/70   )A$ )I.
) ADD 13                                                        )A$ )I.
         LCLA  &INDEX1                                          )A$ )I.
) ADD 123                                                       )A$ )I.
         CLI   TYPEIN,C'X' X COMMAND                            )A$ )I.
         BNE   XPROC1 NO                                        )A$ )I.
         MVC   TYPETRAN(8),=CL8'FGPTYPXX' MAKE IT X OPERATOR COM)A$ )I.
         B     OVERLAY1                                         )A$ )I.
XPROC1   EQU   *                                                )A$ )I.
) ADD 1642                                                      )A$ )I.
         AIF   ('&SUBTASK' EQ 'NO').XPROCX                      )A$ )I.
         TITLE '          ''X''  OPERATOR COMMAND'              )A$ )I.
FGPTYPXX CSECT                                                  )A$ )I.
         USING *,6                                              )A$ )I.
         USING DATAUNIT,2                                       )A$ )I.
         USING COMMONDT,9,8,10                                  )A$ )I.
         USING TIB,7                                            )A$ )I.
         USING JCT,14                                           )A$ )I.
         USING DCT,4                                            )A$ )I.
         OC    TYPEIN+1(20),BLANKS                              )A$ )I.
         CLI   TYPEIN+1,X'40' SPACE NEEDED                      )A$ )I.
         BE    XPROC3 BR IF SPACE                               )A$ )I.
XPROC2   MVC   TYPEIN(L'XPROMSG1),XPROMSG1 BAD COMMAND          )A$ )I.
         MVI   TYPEIN-1,L'XPROMSG1 MSG LENGTH                   )A$ )I.
         B     FUNNELJ GO TYPE ERROR MSG.                       )A$ )I.
```

EXHIBIT 1 cont'd.

For IBM Internal Use Only

```
XPROC3   EQU   *                                                    )A$ )I.
         CLI   TYPEIN+2,C'A' REQUEST FOR ACTIVE TASK                )A$ )I.
         BE    XPROC9 BR IF YES                                     )A$ )I.
         CLC   TYPEIN+2(4),=CL4'LOAD' LOAD                          )A$ )I.
         BE    XPROLOAD BR IF LOAD FUNCTION                         )A$ )I.
         CLC   TYPEIN+2(6),=CL6'REMOVE' REMOVE FUNCTION             )A$ )I.
         BNE   XPROC2 NO, INVALID FUNCTION                          )A$ )I.
         CLI   TYPEIN+8,X'6B' COMMA NEEDED                          )A$ )I.
         BNE   XPROC2 BR IF ERROR                                   )A$ )I.
         CLC   XPROPHAS,TYPEIN+9 IS REQUESTED PHASE ACTIVE          )A$ )I.
         BE    XPROC4 BR IF YES                                     )A$ )I.
         MVC   XPROMSG2+6(8),TYPEIN+9 MOVE PHASE NAME               )A$ )I.
         MVC   TYPEIN(L'XPROMSG2),XPROMSG2 ERR MSG                  )A$ )I.
         MVI   TYPEIN-1,L'XPROMSG2 MSG LENGTH TO ERR MSG            )A$ )I.
         B     FUNNELJ GO TYPE ERROR MSG                            )A$ )I.
XPROC4   EQU   *                                                    )A$ )I.
         MVI   SVCFGX+1,X'F0' MAKE IT UNCONDITIONAL BRANCH AGAIN    )A$ )I.
         MVC   XPROPHAS(8),BLANKS CLEAR PHASE NAME                  )A$ )I.
         L     7,TYPESTAT+4                                         )A$ )I.
         B     TYPE3 DEACTIVATE TYPE TIB                            )A$ )I.
XPROLOAD CLI   TYPEIN+6,X'6B' COMMA NEEDED                          )A$ )I.
         BNE   XPROC2 BR IF NONE                                    )A$ )I.
         CLI   XPROPHAS,C' ' SUBTASK ACTIVE                         )A$ )I.
         BE    XPROC6 BR IF NO SUBTASK ACTIVE                       )A$ )I.
XPROC9   MVC   XPROMSG3+6(8),XPROPHAS MOVE ACTIVE PHASE TO MSG      )A$ )I.
         MVC   TYPEIN(L'XPROMSG3),XPROMSG3 MOVE MSG                 )A$ )I.
         MVI   TYPEIN-1,L'XPROMSG3 MSG LENGTH                       )A$ )I.
         B     FUNNELJ              GO TYPE ERR MSG                 )A$ )I.
XPROC6   EQU   *                                                    )A$ )I.
         LA    1,XPHASA POINT TO 1ST PHASE NAME                     )A$ )I.
XPROC7   CLC   0(8,1),TYPEIN+7 IS PHASENAME VALID                   )A$ )I.
         BE    XPROC8 BR IF YES                                     )A$ )I.
         LA    1,8(1) INCRE TO NEXT PHASE NAME                      )A$ )I.
         CLI   0(1),X'FF' END OF PHASE NAME TABLE                   )A$ )I.
         BNE   XPROC7 BR IF NO                                      )A$ )I.
         MVC   XPROMSG4+6(8),TYPEIN+7 INVALID PHASE NAME TO MSG     )A$ )I.
         MVC   TYPEIN(L'XPROMSG4),XPROMSG4 ERR MSG                  )A$ )I.
         MVI   TYPEIN-1,L'XPROMSG4 MSG LENGTH                       )A$ )I.
         B     FUNNELJ GO TYPE ERR MSG                              )A$ )I.
XPROC8   EQU   *                                                    )A$ )I.
         MVI   SVCFGX+1,X'70' MAKE IT BNE INSTR                     )A$ )I.
         MVC   TMINSTR(4),XPROINST MAKE IF FIRSTIME                 )A$ )I.
         MVC   XPROPHAS(8),TYPEIN+7 SAVE ACTIVE PHASE NAME          )A$ )I.
         B     XPROC9 GO TYPE PHASE NAME                            )A$ )I.
XPROMSG1 DC    CL20'INCORRECT OP COMMAND'                           )A$ )I.
XPROMSG2 DC    C'PHASE           NOT ACTIVE'                        )A$ )I.
XPROMSG3 DC    C'PHASE           NOW ACTIVE'                        )A$ )I.
XPROMSG4 DC    C'PHASE           NOT VALID'                         )A$ )I.
XPHASA   EQU   * PHASE TABLE BEGIN                                  )A$ )I.
&INDEX1  SETA  2 POINT PAST REQUIRED YES OPERAND                    )A$ )I.
.XPROC1  AIF   ('&SUBTASK(&INDEX1)' EQ '').XPROC9 IS IT NULL        )A$ )I.
         DC    CL8'&SUBTASK(&INDEX1)' VALID PHASE NAME              )A$ )I.
&INDEX1  SETA  &INDEX1+1 INCRE TO NEXT ENTRY IN SUBLIST            )A$ )I.
         AGO   .XPROC1                                              )A$ )I.
.XPROC9  ANOP                                                       )A$ )I.
XPHASB   DC    8X'FF' PHASE TABLE END                               )A$ )I.
         LTORG                                                      )A$ )I.
         ORG   FGPTYPXX+1024-8                                      )A$ )I.
         DC    CL8'FGPTYPXX'                                        )A$ )I.
         DROP  6,2                                                  )A$ )I.
.XPROCX  ANOP                                                       )A$ )I.
) END                                                               )A$ )I.
```

EXHIBIT 1 cont'd.

OCR INPUT PREPARATION GUIDE FOR THE 1287/1288

The OCR Input Preparation Guide for the IBM 1287/1288 is now in stock in Mechanicsburg. The form number is GC20-1686-3. This manual contains several sample OCR forms from different application areas including a narrative for the highlights of each form. The manual also contains several human factors considerations for OCR forms design, a step-by-step approach to forms design for the 1287/1288, specifications for forms design for the 1287/1288 and a discussion of all the possible input devices for the 1287/1288 with suggestions and considerations for each of these environments.

INFORMATION ABOUT THE NEWSLETTER

The IBM Installation Newsletter is distributed only to IBM locations and is not available to customers. The Newsletter is separated into two parts. The first part contains information which is intended only for IBM personnel and the pages are labeled For IBM Internal Use Only. THESE PAGES, OR REPRODUCTIONS OF THEM, ARE NOT TO BE GIVEN TO CUSTOMERS. However, using your own good judgement, you may discuss pertinent information from these pages with a customer. The second part contains information which may be reproduced by the Branch Office at their discretion and given to customers. These pages are not labeled with any restrictive classification. Note that when any information on Type III, Type IV, or Prior Use programs is given to a customer, he must be informed that the disclaimers as contained in the Newsletter apply.

YOUR COOPERATION IN PROPERLY HANDLING THE CONTENT OF THE IBM INSTALLATION NEWSLETTER IS NECESSARY FOR ITS CONTINUED PUBLICATION.

Direct inquiries concerning the contents of this Newsletter to:

W. F. Malmborg, Editor
IBM Installation Newsletter
Dept. 812, DPD HQ
112 East Post Road
White Plains, New York 10601
Telephone 914-696-3781
Tie Line 254-3781

SPECIAL PRODUCTS INFORMATION

I. 2501, 2560 Optical Mark Reading Features are now available

II. General Description

Provides the 2501 or 2560 with the ability to read vertical marks made with a number two pencil. The capabilities and applications of these card readers are thus expanded into such areas as meter reading, billing, student registration, order entry, and other markets now requiring mark sensing.

III. Highlights

--Reads marks made by a number two pencil or equivalent.
--Hand marked cards are read directly into the CPU without being reproduced into standard punched cards.
--Marked columns and punched columns may be intermixed within a card.
--Regular punched cards and marked cards may be intermixed.
--Up to 40 columns of marked data or 80 columns of punched data can be read.
--All standard machine functions are still available with optical mark reading installed
--Offers flexibility of card layout.
--Machine thruput is not degraded.
--No special programming is required.
--Optical Mark reading is field installable.

IV. Programming Support

No programming support is provided.

V. Optical Mark Read Input Cards

Special preprinted cards are required. Contact Regional Special Product Marketing for card specifications.

VI. Special Features

1. Switch On/Off Optical Mark Read - Provides an illuminated push button on the operator panel to switch the optical mark read function off. This will allow reading holes in a mark read field, or reading punched cards having printed matter which might activate the mark read function.

2. Eliminate Reading in Rows 11 and 12 - Disables optical mark reading in rows 11 and 12 of the card. This allows the use of pre-printed information in that area of the card such as headers, notes, etc. Optical Reading in the 11 and 12 rows is permanently disabled; however, punched card reading is not affected.

VII. Reference RPQ* Numbers and Area Prices

Name	RPQ No.	MAC	MM Pur.	MM MC
OMR Read Head (2501 and 2560) with one of the following:	841213	112.	4025.	39.
OMR (2501A)	841216	121.	4365.	3.
OMR (2501B)	841217	136.	4900.	3.
OMR (2560)	841218	145.	5240.	3.
Switch On/Off OMR (2501A)	841219	15.	540.	N/C
Swith On/Off OMR (2501B)	841220	15.	540.	N/C
Switch On/Off OMR (2560)	841221	9.	325.	N/C
Eliminate OMR Rows 11 & 12 (2501)	Z00854	N/C	N/C	N/C
Eliminate OMR Rows 11 & 12 (2560)	MA1268	N/C	N/C	N/C

*All RPQ prices shown are "Area" prices. Firm quotations must be obtained from Regional Special Product Marketing before quoting to the customer.

June 19, 1970

IBM INSTALLATION NEWSLETTER

Issue No. 70-12

Table of Contents

TSS/360 RELEASE 7.0 PLANNING INFORMATION

The following information is provided for planning purposes only. The actual availability and specifications of all items will be provided through the official IBM procedures: the Program Announcement and formal documentation.

Version 7 of the Time Sharing System/360 will be available as Type I with Programming Service Classification A. It will provide significant incremental and functional improvements in these areas:

Functional Improvements

Remote Job Entry (RJE) has been incorporated; supporting a 2780 Type I terminal, it permits its user to initiate any batch or non-conversational TSS/360 job .... PL/I-F, Version 5 has been programmed to operate in the TSS/360 environment; PL/I programs can be compiled and executed on TSS/360.

Performance

Many system changes have been made to improve performance by reducing paging overhead .... Device management has been changed to improve performance .... TSS/360 now provides selective loading of particular functions at STARTUP time.

Reliability and Serviceability

Addition of the facility to permit or prevent the creation of multivolume data sets .... improved VAM interlocking facility .... additional On Line Tests .... improved I/O outboard error recordings .... new ability to prevent tasks from entering the system when no auxiliary space is available .... adds a new alternate I/O path retry method.

Human Factors

Addition of the Hex Option in Text Editor .... improved keyboard/cardboard origin character and data set compression .... improved System Messages for standardization and clarity .... commands changed for consistency and improved operation.

1. Remote Job Entry (P69-48)

Remote Job Entry (RJE) has been incorporated into the system. This function supports a 2780 Type I terminal that is attached via a 2701 Data Adapter Unit and permits the user to initiate any batch or nonconversational job from this terminal. All outputs created by the remote job are returned to the originating 2780 unless otherwise directed. The remote job function interfaces with the system as an extension of the current BULK I/O facility. RJE will operate with a 2780 that is equipped with or without the Multiple Record Transmission feature. The 2780 will operate through a 2703 Data Adapter Unit with modifications introduced at SYSGEN time. On Line Test facilities for the 2701 and the 2780 are not provided in this release.

2. PL/I (P69-48)

PL/I, Level-F, Version 5 has been programmed to operate in the TSS/360 environment. PL/I programs can be compiled and executed on TSS/360. It will operate in the 24 bit addressing mode and be compatible with OS PL/I-F except for SORT/MERGE, multitasking, checkpoint and the regional I/O and QTAM facilities of the I/O environment option. For running PL/I, an option is available through the use of a new LOGON parameter to limit the Virtual Memory address to 24 bits when running with 32 bit hardware.

3. Selective Load

TSS now provides the users with the ability to selectively load or not load modules associated with certain functions. At STARTUP time, by specifying the functions to be deleted, the related control sections will automatically be eliminated from the loading and linking process.

4. On-Line Test Improvement

ON-LINE Tests are included to support the 1403, 2400, 2540, 2541, and 2703.

5. New Commands

Two related commands, DDNAME? and JOBLIBS, are added to the Command System. DDNAME? displays either all of the user's active DDNAMEs and DSNAMEs or just those in the JOBLIB chain. JOBLIBS moves an active JOBLIB into the top position of the program library list; with this command, the existing last-in first-out JOBLIB order can be overriden.

PRMPT is a new command that helps the user retrieve messages and message explanations with provision for the insertion of variable information in messages that are written to allow such insertion.

6. Password Security

If LOGON is entered without a required password, the system issues an "ENTER PASSWORD" prompt, and an overtyped line is provided following the prompt. The user then types his password over this overtyped line, making the password harder to read.

7. System Messages

System messages in the SYSMLF data set have been reworked in an attempt to obtain improved consistency and relevancy of meaning. Problems may still exist in individual messages and these should be reported to IBM. However the overall effect should be an improvement.

The main emphasis was in the text of the messages themselves. In general, what the message is trying to say should be more clear or concise, and more relevant to the type of person who is receiving the message. Standard words and phrases were used in the text of the message and these have a global definition which may be obtained by using the explain word option of the EXPLAIN command.

Hints for using the System Message Facilities

a) The content of the standard messages has been reworked to apply only to normal users (i.e. not system programmers). Those who want additional information concerning the internal operation of the system should use the EXPLAIN command to obtain whatever information of this type is available.

b) In situations where the user is expected to respond to the system by entering one of several options, the user may find that the options are not described in the standard message. The user should then use the explain response option of the EXPLAIN command for a complete description of the options available to him as well as the specific value which he is expected to enter.

c) The filter codes which were assigned to individual messages have been restructured so that the average user should normally be able to run with the warning filter code. He obtains this by stating, 'DEFAULT LIMEN = W'. The novice user should use 'LIMEN = I'. If the user wishes to see only messages which describe abend situations or messages which require a response in order for processing to continue he may run with LIMEN = T. All other messages will be filtered out.

e) The new system default for brevity is BREVITY = T. This means that unless the user takes other action he will only see the text of all standard messages. If the user wishes to see both the text and the message id he must use BREVITY = S. It is suggested that BREVITY = S be used when a terminal sheet is created for submission with an APAR, since the message id may be of value to system programmers who are trying to reconstruct the circumstances of the error.

DELETED TYPE III PROGRAMS

The following Type III programs will be deleted by the Program Information Department on July 19, 1970. Until that date, orders for these programs will be accepted by PID. These deletions have been requested by the authors.

360D-00.1.019	AN OS/360 MULTI-PURPOSE UTILITY FOR MULTI-PROGRAMMING
360D-03.3.008	COURSEWRITER III/OS A REAL TIME COMPUTER ASSISTED INSTRUCTIONAL SYSTEM