Systems

A Guide to the System/370 Model 135

IBM

Systems

A Guide to the System/370 Model 135

This guide presents hardware, programming systems, and other pertinent information about the IBM System/370 Model 135 that describes its significant new features and advantages. The contents are intended to acquaint the reader with the Model 135 and to be of benefit in planning for its installation.

IBM

Second Edition (March 1971)

This guide is intended for planning purposes only. It will be updated from time to time to reflect system changes; however, the reader should remember that the authoritative sources of system information are the Systems Reference Library (SRL) publications for the Model 135, its associated components and its programming support. These publications will first reflect such changes.

Copies of this and other IBM publications can be obtained through IBM branch offices.

A form has been provided at the back of this publication for readers' comments. If this form has been removed, address comments to: IBM Corporation, Technical Publications Department, 1133 Westchester Avenue, White Plains, New York 10604.

PREFACE

It is assumed that the reader of this document is familiar with System/360. The reader should have a general knowledge of System/360 architecture, channels, I/O devices, and programming systems support. This guide highlights only those Model 135 hardware, I/O, and programming systems features that are different from those of System/360 models and discusses their significance. This publication applies to systems with 60-cycle power. Additional, more detailed information regarding System/370 Model 135 hardware and programming systems support can be found in the following SRL publications:

IBM System/370 Model 135 Functional Characteristics (GA33-3005)

IBM System/370 Model 135 Configurator (GA33-3006)

IBM System/370 Installation Manual - Physical Planning (GC22-7004)

IBM System/370 System Summary (GA22-7001)

IBM System/370 I/O Configurator (GA22-7002)

IBM System/370 Principles of Operation (GA22-7000)

IBM 2319 Disk Storage Component Summary (GA26-1606)

IBM 3215 Console Printer-Keyboard Model 1 Component Description (GA24-3550)

Component Summary: 3830 Storage Control, 3330 Disk Storage (GA26-1592)

3211 Printer and 3811 Control Unit Component Description (GA24-3543)

IBM Component Description: 3803/3420 Magnetic Tape Subsystem (GA32-0020)

IBM Component Description: 3505 Card Reader and 3525 Card Punch (GA21-9124)

Form-Design Considerations - System Printers (GA24-3488)

Emulating the IBM 1401, 1440, and 1460 on IBM System/370 Models 135, 145, and 155 using OS/360 (GC27-6945)

IBM System/360 Operating System:

- Planning for the IBM 3211 Printer Data Management Macro Instructions and Services (GC21-5008)
- Program Planning Guide for the DOS Emulator on IBM System/370 Models 135, 145, and 155 (GC24-5076)
- Planning for the IBM 3505 Card Reader and the IBM 3525 Card Punch Data Management Macro Instructions and Services (GC21-5027)

Emulating the IBM 1401, 1440, and 1460 on IBM System/370 Models 135, 145, and 155 using DOS/360 (GC33-2004)

IBM System/360 Disk Operating System:

- IBM 3211 Printer Program Planning Guide (GC24-5085)
- Program Planning Guide for MCAR/CCH Function for IBM System/370 Model 135 (GC24-5089)
- Planning for the IBM 3505 Card Reader and IBM 3525 Card Punch (GC21-5034)
- Planning Guide for Programming the 3330 Direct Access Storage Facility (GC33-5004)

CONTENTS

FIGURES

TABLES

SECTION 01: SYSTEM HIGHLIGHTS

The System/370 Model 135 is designed to enhance, extend, and broaden the successful concepts of System/360 architecture. It is a general purpose growth system for System/360 Model 25 and Model 30 users that provides significant price performance improvement without the necessity of major reprogramming. The System/370 Model 135 retains and extends the range of commercial and scientific data processing capabilities offered by System/360 Models 25 and 30. It is compatible with System/370 Models 145, 155, and 165.

Transition from System/360 Models 25 and 30 to the System/370 Model 135 can be accomplished with a minimum of effort because most current System/360 user programs, I/O devices, and programming systems are upward compatible with the new system. Additional capabilities will be added to DOS and OS MFT to support new features of the Model 135, thereby providing proven operating system performance as well as continuity.

Transition with little or no reprogramming is also provided for 1401/1440/1460 users who are presently emulating on System/360. Improved emulators for these systems that operate under DOS or OS MFT control on the Model 135 will be available.

DOS users who wish to install OS on their Model 135 can ease the transition by using the standard OS/DOS Compatibility feature. An OS DOS Emulator program is provided that supports emulation of a DOS multiprogramming system under OS MFT control on the Model 135.

Highlights of the Model 135 are as follows:

- Upward compatibility with most System/360 architecture and programming has been maintained.

- Internal performance is two to four and one-half times that of the Model 30 for a typical commercial instruction mix and three and one-half to seven times that of the Model 30 for a typical scientific instruction mix.

- The following are CPU features of the Model 135.

  The Model 135 standard instruction set includes new general purpose instructions in addition to the powerful System/360 instruction set. These instructions enhance decimal arithmetic performance, simplify the handling of nonword-size data that is processed using registers, eliminate the need for multiple move or compare instructions or move subroutines, and facilitate record blocking and deblocking, field padding, and storage clearing.

  An extended precision floating-point option is available in addition to the no-charge floating-point arithmetic option. Precision of up to 28 hexadecimal digits, equal to up to 34 decimal digits, is provided by the extended precision data format.

  An interval timer of 3.3 milliseconds resolution to improve job accounting accuracy is standard. (A 16.6-ms resolution timer is available for Models 25 and 30.)

  A time of day clock is included as a standard feature to provide more accurate time of day values than does the interval timer. The clock has a 1-microsecond resolution.

Detection of CPU errors during instruction execution causes most instructions to be retried automatically by hardware, without programming assistance.

- Functionally improved relocatable emulators are available that operate under operating system control. Concurrent execution of System/370 programs with any combination of 1401, 1440, and 1460 programs in a multiprogramming environment is supported. The 1401/1440/1460 Compatibility feature is a no-charge option.

- The standard OS/DOS Compatibility feature permits emulation of a DOS system under OS concurrently with execution of other OS jobs. Both DOS and 1400 emulation can operate together on a Model 135 under OS MFT control if enough storage is available.

- Operator console devices with alter/display mode, announced for Models 145 and 155, are available.

  The 15-cps 3210 Model 1 Console Printer-Keyboard

  The 85-cps 3215 Console Printer-Keyboard

- The following channel and I/O adapter features are available for the Model 135.

  One or two high-speed selector channels and an integrated adapter for 2314-type direct access devices can be attached in addition to the standard byte multiplexer channel. The latter can have 16 or 64 subchannels. The first selector channel installed can operate at a maximum data rate of 1.3 megabytes (MB). The second selector can handle data rates up to 1.2 MB. Therefore, the Model 135 can support significantly faster I/O devices than can Models 25 and 30.

  The optional Integrated File Adapter (IFA) feature allows lower cost, direct attachment of 2314-type disk drives to a Model 135. Installation of a selector channel and a disk control unit are not required. The 2319 Disk Storage Model A1 unit announced with the Model 145 (three drives with a total maximum capacity of up to 87 million bytes) and either 2312 Disk Storage (one drive) or 2318 Disk Storage (two drives) can be connected via the adapter for a configuration of 3, 4, or 5 natively attached disk drives. Facilities consisting of from one to eight 2314-type drives, plus a spare, can also be attached to one selector channel when the IFA is installed.

  The optional Integrated Communications Adapter (ICA) for the Model 135 offers lower cost communication line attachment than is currently available for Models 25 and 30. The ICA permits attachment of up to eight low- and medium-speed lines, which can handle any combination of start/stop, alphanumeric display, and binary synchronous communications. It supports most of the same terminals as the 2701 Data Adapter Unit and the Model 25 ICA. The following can be attached to a Model 135 using the ICA: the 1050 Data Communication System, 2740 and 2741 Communication Terminals, System/7, 2260 and 2265 Display Stations (via 2848 and 2845 Display Control), and the binary synchronous terminals supported by the 2701. Communication lines can also be attached to the Model 135 via 2701, 2702, and 2703 transmission control units.

  Block multiplexer mode of operation for all installed selector channels is a no-charge optional feature. A block multiplexer channel is a superset of a selector channel. When used in conjunction with rotational position sensing devices, block

multiplexer channels can increase total system throughput by permitting increased amounts of data to enter and leave the system in a given time period. A single block multiplexer channel can support interleaved, concurrent execution of multiple high-speed I/O operations.

Block multiplexer channels support attachment to the Model 135 of the 3330 direct access facility, which cannot be attached to Models 25 and 30.

Channel retry data is provided when channel errors occur, so that error recovery routines can retry I/O operations.

- The following significant storage features are provided by the Model 135.

  All system storage--local, control, and processor (main)--is implemented using monolithic technology instead of discrete ferrite cores.

  96K to 240K of monolithic processor (main) storage is available--almost four times the maximum main storage available on the Model 30. Processor storage has a read cycle time of 770 nanoseconds and a write cycle time of 935 nanoseconds for two bytes (includes fetch time for the next microinstruction).

  Reloadable, monolithic control storage is used to contain the microcode necessary for system operation. Control storage contains the microcode required for standard and optional features and can be expanded from 24K to 48K in two 12K increments. Use of writable, instead of read-only, control storage offers the advantages of system cost reduction and improved system serviceability.

  Byte boundary alignment is permitted for the operands of non-privileged instructions to eliminate the necessity of adding padding bytes within records or to blocked records for the purpose of aligning fixed- or floating-point data.

  Error checking and correction (ECC) hardware, which automatically corrects all single-bit processor and control storage errors and detects all double-bit and some multiple-bit errors, is standard.

- I/O devices include the following.

  Most currently announced I/O devices for System/360 Models 25 and 30 can be attached.

  The new 3505 Card Reader and the 3525 Card Punch with optional card read capability can be attached. A variety of models are available. They offer 80-column card users configuration flexibility, new functions, high reliability, and greatly expanded error recovery facilities.

  Models B1 and B2 of the 3505 Card Reader can operate at 800 and 1200 cards per minute, respectively. Significant new optional features include Optical Mark Reading and Read Column Eliminate. The latter is designed to permit the successful reading of cards containing internal perforations or other holes that normally would cause an error.

  Models P1, P2, and P3 of the 3525 Card Punch can punch and, optionally, read 100, 200, and 300 cards per minute, respectively. New features of this unit include automatic punch retry when an error is detected during non-read/punch operations

(standard) and optional card printing. A two-line print feature and a multiline print feature (up to 25 lines) are available.

The high-speed 3211 Printer, with a tapeless carriage and print speed of 2000 alphameric lines per minute, is attachable. The tapeless carriage decreases operator intervention by eliminating carriage tape loading and unloading.

The 3803/3420 Magnetic Tape Subsystem is attachable. Models 3, 5, and 7 of the 3420 Magnetic Tape Unit, with data rates of 120 KB, 200 KB, and 320 KB, respectively, at 1600-BPI recording density, are provided. Phase-encoded recording, which automatically corrects all single-bit read errors in-flight, is used. This new tape subsystem offers improved price performance; Dual Density and Seven-Track features for compatibility with, and conversion of, 2400-series tape volumes; greatly reduced operator handling through implementation of such features as automatic tape threading and cartridge loading; lower cost tape-switching than is currently provided; and enhanced reliability, availability, and serviceability features.

The 3330 Disk Storage Facility can be attached. This facility offers significantly faster seeks and more than twice the data rate of the 2314 facility, more than three times the capacity of the 2314, automatic error correction features, and the new rotational position sensing and multiple requesting capabilities.

The 3330 has an 806-KB data transfer rate, average seek time of 30 ms, and full rotation time of 16.7 ms. A two-, four-, six-, or eight-drive facility can be configured. Each drive has a maximum capacity of 100 million bytes.

- Extensive hardware and programming systems error recovery and repair features are provided to enhance system reliability, availability, and serviceability.

As the highlights indicate, Model 25 and 30 users now have a broader range of Model 135 configurations from which to choose when tailoring a growth system with improved throughput and expanded capabilities.

Specifically, the Model 135 offers the following advantages when compared to Models 25 and 30.

Larger, Faster Processor (Main) Storage Sizes

Processor storage sizes of 96K, 144K, 192K, and 240K bytes are provided. The Model 25 can have a maximum of 48K, while 64K is the largest main storage size provided by a Model 30. The cycle time of Model 135 processor storage is almost two and one-half times faster than that of the Model 30, during which two bytes instead of one can be accessed. This improved cycle time increases internal performance and permits faster I/O devices to be attached to the system.

Additional storage can contribute significantly to system performance and capabilities. Specifically, the addition of more processor storage provides the Model 135 user with the ability to:

- Execute more or larger jobs concurrently, including new application and integrated emulator jobs
- Add and expand applications, such as graphics, teleprocessing, time-sharing, and data-based, that require larger amounts of storage
- Use higher level language translators and linkage editors that provide more functions and execute faster

- Execute larger processing programs without the necessity of overlay structures
- Allocate more storage to language translators and sorts to improve their execution speed
- Use more and larger I/O areas to speed up input/output operations and optimize use of direct access storage and tape media space
- Include system generation options that improve control program performance and support additional functions

Greatly Expanded I/O Capabilities

The fast internal performance of the Model 135, together with expanded use of multiprogramming, requires that more data be available faster than on the Model 30. A variety of channel options is provided.

The Model 135 supports more and faster concurrent high-speed channel operations than Models 25 and 30 and block multiplexer channels, which are not provided for these System/360 models. Integrated I/O adapters, not available for the Model 30, are also offered.

The I/O features of the Model 135 provide:

- Lower cost direct attachment of 2314-type disk storage drives (2319-A1 and 2312 or 2318) via the Integrated File Adapter and channel attachment of 2319-B disk storage
- Lower cost direct attachment of up to eight communication lines with start/stop, alphanumeric display, and binary synchronous terminals via the Integrated Communications Adapter
- Attachment of new high-speed direct access devices such as the 3330 facility
- Attachment of the 3505 reader and 3525 punch, not attachable to Models 25 and 30
- Potential increases in channel throughput via use of block multiplexing with rotational position sensing to improve effective data transfer rates
- A significantly higher attainable aggregate channel data rate than the Model 30 to balance the higher performance capabilities of the Model 135 CPU

Faster I/O Devices and Increased Direct Access Storage Capacity

The Model 135 supports faster magnetic tape units than do Models 25 and 30. The 3420 Model 7, with a data rate of 320 KB, cannot be attached to the Model 30. A tape unit faster than 60 KB cannot be attached to the Model 25.

A Model 135 I/O configuration can also include significantly more and faster direct access storage. For example, the Model 30 is limited to having 2314 facilities on only one channel, while the 2314 is not a standard device for the Model 25. The Model 135 can have 2314 facilities on both selector channels if the IFA is not present, or on the IFA and one selector channel. In addition, the 3330 facility can be attached to a Model 135 and it provides considerably more capacity and faster data access than 2314 facilities because of higher data transfer rates, faster rotation, and new features. Rotational position sensing and multiple requesting used with block multiplexing can improve I/O throughput by making more efficient use of channel

time. The 3330 direct access facility also offers higher availability through use of new hardware-only and program-assisted error correction features.

The 3330 facility provides large capacity and fast access for less cost per bit. It is a growth device that offers improved price performance for the 2314 facility and the 2321 Data Cell Drive. Like 2314 disk storage, the 3330 facility is designed to be used in every area in which direct access storage is needed, for example:

- As a system residence device and for program library storage
- In teleprocessing applications for message queuing and for residence of online applications data
- In online, data-based applications, such as management information systems
- In time-sharing (or interactive) environments as swap devices and for online work storage (for program and data residence)
- As high-speed work storage for sorting, assembling, and link editing
- For residence of data indices, such as for ISAM data sets

Summary

The combination of new and enhanced hardware, availability, and input/output facilities, expanded operating system support, integrated 1400 emulation, and DOS emulation under OS MFT provided by the Model 135 offers Model 25 and 30 users expanded computing capabilities without the necessity of a large conversion effort. Little or no time need be spent modifying operational System/360 code or programs currently being emulated. Users of 1400 systems can upgrade directly to a Model 135 and an operating system environment with a minimum of reprogramming, and DOS users can convert to OS more easily because of the availability of DOS emulation. Existing CPU-bound programs can execute faster because of the increased internal performance of the Model 135, while I/O-bound programs can benefit from the use of more storage, more channels, faster I/O devices, and block multiplexing. The Model 135 also offers economic and flexible entry into communications-based applications.

The increased power and new functions of the Model 135 provide the base for expanded applications growth and penetration of previously marginal application areas. The improved price performance of the Model 135 offers the user the opportunity to widen his data processing base for less cost than was previously possible.

SECTION 10: ARCHITECTURE, TECHNOLOGY, AND SYSTEM COMPONENTS

10:05 ARCHITECTURE DESIGN AND SYSTEM TECHNOLOGY

ARCHITECTURE DESIGN

The basic design objectives embodied in System/370 Model 135 architecture provide System/360 and 1401/1440/1460 emulator users with a growth system in the intermediate system range that incorporates improvements and additions to System/360 architecture. The Model 135 provides new system capabilities, performance improvements, and features to enhance system reliability, availability, and serviceability. This has been achieved under the following conditions:

- System/370 Model 135 architecture is upward compatible with that of System/360 models so that most user programs written for System/360 will run efficiently on the Model 135 without modification.
- Programming systems support for the Model 135 is based on certain programming currently provided for System/360 models, namely, DOS and OS MFT.
- Most currently announced System/360 and System/370 I/O devices will operate on the Model 135. (See Section 20:05 for a list of the I/O devices that cannot be included in a Model 135 configuration.)
- The open-ended design characteristic of System/360 has been preserved and extended on System/370.

As a result of the architecture design criteria used for this new system, all programs written for System/360 (Models 25 and up) will operate on a System/370 Model 135 with a comparable hardware configuration, with the following exceptions:

1. Time-dependent programs. (They may or may not execute correctly.)
2. Programs using machine-dependent data such as that which is logged in the machine-dependent logout area. (DOS MCRR and OS SER error-logging routines for System/360 models will not execute correctly.)
3. Programs that use the ASCII mode bit in the PSW. (ASCII mode is not implemented.)
4. Programs that depend on the nonusable lower processor storage area being smaller than 512 bytes.
5. Programs deliberately written to cause certain program checks.
6. Programs that depend on devices or architecture not implemented in the Model 135, for example, the native file of the Model 44, relocation implemented in the Model 67, etc.
7. Programs that use model-dependent operations of the System/370 Model 135 that are not necessarily compatible with the same operations on System/360 models.
8. Programs that depend on the validity of storage data after system power has been turned off and then on.

Note that these are the same types of implementation-dependent restrictions that exist for compatibility among System/360 models.

SYSTEM TECHNOLOGY

The Model 135 uses monolithic system technology (MST) for logic circuitry, as do System/370 Models 145, 155, and 165. In addition, the Model 135, like the Model 145, embodies a significant technological advancement in the area of system storage implementation. That is, processor storage, as well as control and local storage, is implemented using monolithic technology instead of wired, discrete ferrite cores. Models 135 and 145 are two of the first IBM systems that use monolithic storage exclusively.

Monolithic storage is similar in design to monolithic logic circuitry, the latter representing a technological advancement over the solid logic technology (SLT) introduced with the announcement of System/360. Since the technology associated with monolithic storage is like that used to produce monolithic logic, monolithic storage can be batch-fabricated.

Solid Logic Technology (SLT)

Monolithic technology is a breakaway from the hybrid circuit design concept of SLT and can best be explained by comparison with SLT. As shown in Figure 10.05.1, SLT circuits were implemented on half-inch ceramic squares called substrates. Metallic lands on the substrate formed interconnections onto which the components were soldered. These components consisted of transistors and diodes, which were integrated on silicon chips about the size of a pinhead, and thin film resistors. An SLT chip usually contained one component, and several chips and resistors were needed to form a circuit. In general, an SLT substrate contained a single circuit.

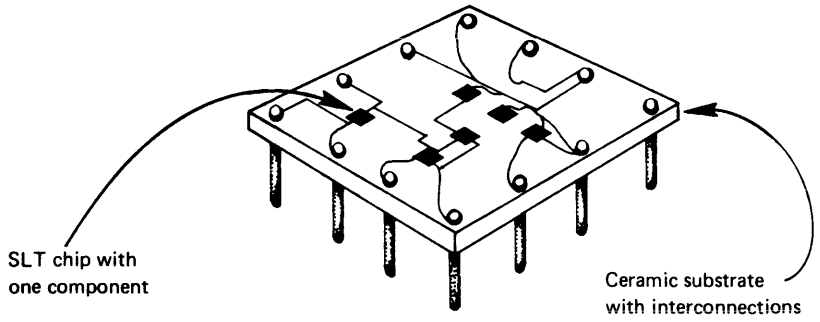


Figure 10.05.1. SLT substrate

Monolithic System Technology (MST)

Monolithic system technology also makes use of a half-inch-square ceramic substrate with metal interconnections onto which chips are placed. However, in monolithic logic circuitry, large numbers of elementary components, transistors, resistors, etc., are integrated on a single chip. In the Model 135, an MST logic chip is slightly over a sixteenth of an inch square and contains over 100 components, which can form up to eight interconnected circuits. This compares to a single component on an SLT chip. MST logic modules, each

consisting of one substrate, are mounted on circuit cards, which are in turn mounted on circuit boards (as in SLT logic).

MST logic offers the following advantages over SLT:

- MST logic circuitry is intrinsically more reliable because many circuit connections are made on the chip, significantly reducing the number of external connections.
- Faster circuit speeds can be obtained because the distance between circuits is considerably shorter. For example, the MST circuits in the Model 135 are about four times as fast as the SLT circuits in the Model 30.
- Space requirements for logic circuitry are reduced by the significantly higher density of components per chip.

Monolithic Storage

Monolithic storage design incorporates the same concepts described for monolithic logic. However, storage bits instead of logic circuits are implemented on a chip. In the Model 135, a monolithic storage array chip is approximately an eighth of an inch square and contains a little more than 1400 components, or about 174 interconnected circuits. These circuits form storage bits and support circuitry on the chip. In the Model 135, one monolithic storage array chip contains 128 storage bits and their associated decoding, addressing, and sensing circuitry.

As shown in Figure 10.05.2, two storage array chips are mounted on a half-inch-square substrate, and a pair of substrates is packaged into a storage array module. Each half-inch-square storage module contains 512 storage bits and is mounted on a storage array card. In outward appearance, therefore, monolithic storage looks like monolithic logic circuitry.

Storage array cards are packaged into basic storage modules (BSM). In the Model 135, a processor storage BSM contains 48K bytes of processor storage and its associated circuitry. The storage cards used, illustrated in Figure 10.05.3, are approximately 3-1/2 by 4-3/4 inches in size and contain 12K storage bits. The 48K-byte BSM is shown in Figure 10.05.4. It is approximately 13-1/4 inches long, 5-1/2 inches deep, and 9 inches wide. A 96K Model 135, for example, contains two basic storage modules of processor storage.

Since power is required to maintain a one or zero state in a monolithic storage bit, data is lost when power is turned off, and monolithic storage is therefore said to be volatile. This is not true of core storage, which retains a magnetized state when power is removed.

The following are the advantages of monolithic over core storage:

- Faster storage speeds can be obtained, first, because of the shorter distances between storage circuitry and, second, because of the nondestructive read-out capability of monolithic storage. Since core storage read-out is destructive, a regeneration cycle is required after a read and is also used prior to a write. Regeneration cycles are not required for monolithic storage.
- Storage serviceability is enhanced because storage is implemented in accessible, easily replaceable cards, each of which is a complete functional storage component. Diagnostic routines can be written that need only identify the failing storage card (as opposed to

the specific storage element), which can be replaced in a matter of minutes. Storage increments can also be field installed rapidly.

- Space requirements for system storage are reduced. Dense bit packaging per chip is achieved by the use of monolithic technology and by the fact that the regularity of a storage pattern lends itself to such packaging. For example, 96K of monolithic processor storage on the Model 135 requires half the amount of space as does 64K of core storage on the Model 30.

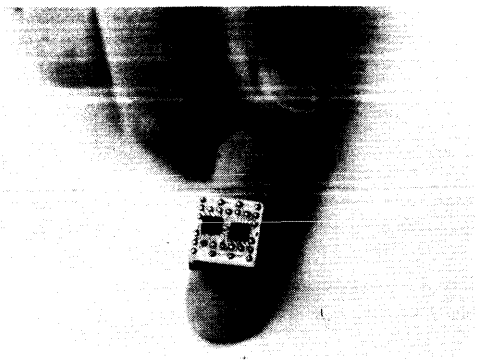


Figure 10.05.2. Monolithic storage array module containing 512 bits

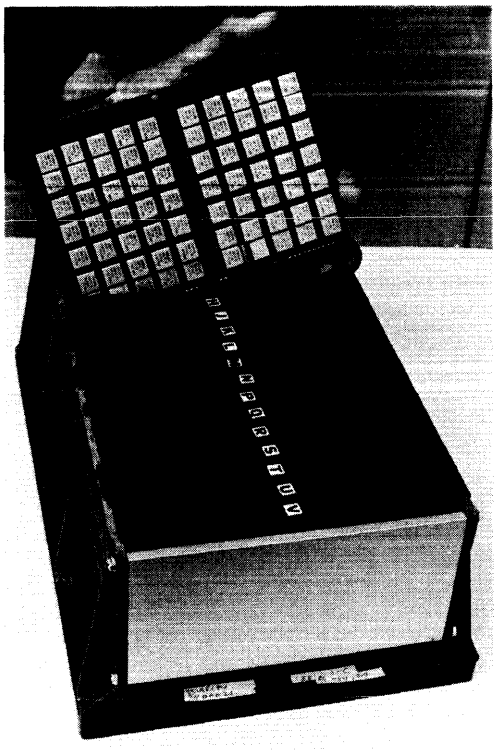


Figure 10.05.4. Model 135 basic storage module containing 48K bytes of monolithic storage

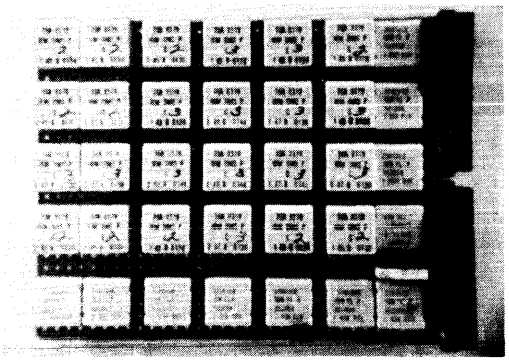


Figure 10.05.3. Monolithic storage array card containing 12K bits

MAJOR COMPONENTS

A physical layout of the Model 135, including natively attached direct access devices, is shown in Figure 10.05.5. The major sections of the Model 135 computing system are the processor (CPU), storage, channels, the system control panel and console, and the console file. All CPU logic, storage, channels, and I/O adapters are contained within the CPU frame. A motor generator set, external to the CPU frame, is used to supply power to the system. Each component and its new features are discussed in the subsections that follow. Programming systems support of these new features is covered in Section 30. Reliability, availability, and serviceability (RAS) hardware features are mentioned only briefly. A full discussion of both hardware and programming systems RAS facilities is contained in Section 50.

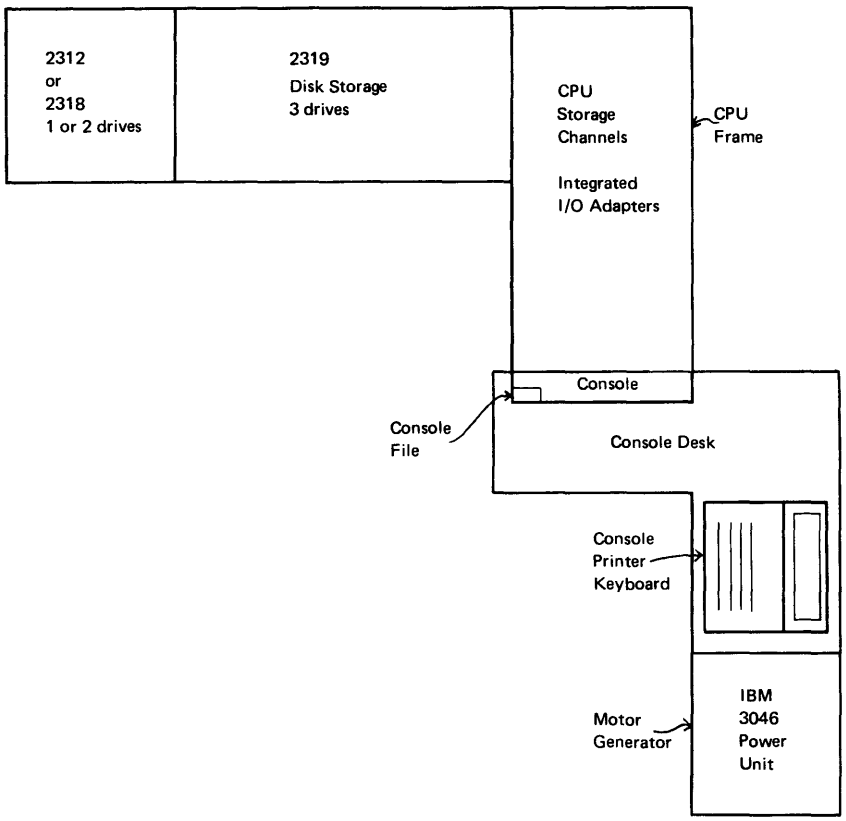


Figure 10.05.5. Model 135 physical layout (drawn to scale)

A motor generator set (MG set) is used to supply power to the Model 135. Use of an MG set reduces the size and cost of power supplies contained in the main frame. Use of a motor generator for the CPU power supply is of benefit when small reductions occur in the line voltage supplied to an installation.

The stand-alone IBM 3046 Power Unit Model 1, which contains a motor generator, is required to supply system power for the Model 135. This air-cooled unit takes the 60-Hz (cycle) power supplied by the building

electrical distribution system, converts it to 400-Hz power, and returns it to the power unit in the CPU, which acts as a distribution unit for power in the CPU.

The IBM 3046 is 24 inches wide, 45 inches long, and 29 inches high. It can be located anywhere up to 25 feet away from the CPU. As shown in Figure 10.05.5, it can be placed next to the console printer-keyboard for use as an additional work area, as it is the same height as the console. The side covers of the 3046 are interchangeable. Only one cover has exhaust ports for the cooling air. Wherever the unit is placed, warm air exhausting from the unit can be directed away from operator working areas.

10:10 THE CENTRAL PROCESSING UNIT (CPU)

The central processing unit contains all the elements necessary to decode and execute the instructions in the System/370 Model 135 instruction set and, optionally, those in the compatibility feature required by the 1401/1440/1460 emulator programs. All CPU functions and most channel operations are controlled by the microprogram contained in the two-byte (18-bit) control words in reloadable control storage (RCS), which is housed in the CPU (discussed in Section 10:15).

The Model 135 has a variable-length CPU cycle time. Cycle times vary from a minimum of 275 to a maximum of 1430 nanoseconds, in 165-nanosecond increments, depending on the operation performed. The data path in the CPU is two bytes wide. If the two bytes fetched by the CPU are located on a doubleword boundary in processor storage, the adjacent two bytes can be fetched, if needed, in an additional 165 nanoseconds.

Monolithic local storage is used as an intermediate storage area and is shared by the CPU, the integrated channels, and the integrated I/O adapters. Local storage contains the general purpose and floating-point registers as well as work areas (called zones) for the CPU, the channels, and the I/O adapters. Only one work zone can be accessed at a time.

Extensive parity checking is done in the CPU to ensure the validity of the data being used. Data paths within and to the CPU are parity-checked, as are microcode words and adder sums. Automatic retry of most instructions that fail because of CPU errors, without programming assistance, is provided as an availability feature and is discussed in the RAS section.

The program states in which the Model 135 is operating are reflected in the current program status word (PSW) and in new CPU status indicators, called control registers, which are contained in control storage. Up to 16 control registers, 0-15, can be addressed; however, only two (0 and 14) are implemented in the Model 135. They are program addressable when the CPU is in the supervisor state. A control register can be set with the new LOAD CONTROL instruction, and its contents can be placed in processor storage with the new STORE CONTROL instruction. Additional status indicators contained in control registers are required in order to support new system functions. Logically, a control register is 32 bits in size.

The contents, layout, and function of fixed locations 0-127 in System/370 models are identical to these locations in System/360 models with one exception. Bit 12 in the PSW, which sets EBCDIC or ASCII mode in System/360 models, is not used for this purpose in the Model 135 and must be set to zero. ASCII mode is not implemented in the Model 135,

nor was the mode bit supported by IBM programming systems provided for System/360 models, as the expectation that System/360 USASCII-8 would become the ASCII standard has not been borne out.

However, ASCII-encoded tapes will be supported by certain DOS and OS programs. That is, ASCII mode tapes will be accepted as input and converted to EBCDIC for processing. The capability of writing ASCII mode tapes is also provided. (See Section 30 for a discussion of DOS and OS support of ASCII mode tapes.)

To enhance system availability and serviceability, the implementation of the machine check level of interrupt on the Model 135 has been considerably altered from its implementation in Models 25 and 30 (see Section 50). However, the other four interrupt levels operate in the same manner on Models 25, 30, and 135 except for the expansion of external interrupt masking in the Model 135. Three external subclass mask bits, which allow selective masking of external signals (2-7), interval timer, and console panel interrupt key interrupts, are provided in control register 0. When the PSW external mask bit is off, all three external interrupt types are disabled. When the PSW external mask bit is on, a console interrupt key, an interval timer, or an external signal interrupt occurs only if its associated subclass mask bit is on also.

Significant new features of the Model 135 CPU are as follows.

EXPANDED INSTRUCTION SET

The standard instruction set for the System/370 Model 135 is a superset of that provided for System/360 Models 25 and 30. It consists of the System/360 instruction set plus new instructions that support System/370 architecture and provide additional functions. The Model 135 standard instruction set includes all general purpose and I/O instructions and all binary and decimal arithmetic instructions. Storage protect and time of day clock instructions are also standard. The new STORE CPU ID instruction permits a program to determine the model upon which it is operating and provides the system serial number. The new STORE CHANNEL ID instruction can be used to identify the types of channels present in the system. Other new instructions are:

- General purpose instructions

  Six general purpose instructions, which will be of benefit to both control and processing program performance, have been added to the Model 135 standard instruction set.

  SHIFT AND ROUND DECIMAL provides right or left shifting of packed decimal data using a single instruction. This instruction can save 6 to 18 bytes of instruction storage and instruction execution time for each decimal shift and round operation performed in commercial processing.

  MOVE LONG provides for the movement of up to 16 million bytes from one location in storage to another with a single instruction, thereby removing the current limitation of 256 bytes per move. A check for the possibility of destructive overlap is made by the hardware prior to the movement of any data and the MOVE LONG instruction is not executed if operand destruction can occur. This instruction can eliminate the necessity of multiple move instructions or the inclusion of move subroutines. The format and operation of MOVE LONG facilitates efficient record blocking and deblocking, field padding, and storage clearing, frequently performed operations in commercial processing. The new COMPARE

LOGICAL LONG instruction can be used to compare logically two fields of up to 16 million bytes in length, thus removing the current 256-byte limit on byte compares. In addition, when an unequal compare occurs, the two characters that caused the inequality are identified.

The MOVE LONG and COMPARE LOGICAL LONG instructions are interruptable. Thus, when an I/O operation terminates during their execution, the interrupt is taken and the channel is not held up awaiting termination of what might be a lengthy move or compare.

COMPARE LOGICAL, INSERT, and STORE CHARACTERS UNDER MASK instructions provide byte addressability within the general registers and permit nonword-size data that is not on a word boundary to be compared to data in a register, loaded into a register, and stored from a register. These three instructions can be of most benefit to control program programmers, to compiler writers, and to others who must manipulate processor storage addresses.

- Extended Precision Floating Point

  The optional extended precision floating-point feature requires the floating-point arithmetic option as a prerequisite. The latter is a no-charge feature. Extended precision is provided for use in application areas in which the precision provided by the long-form floating-point format is not large enough.

  Precision of up to 28 hexadecimal digits, equal to up to 34 decimal digits, is provided by the extended precision data format. Extended precision is achieved by using two doublewords (16 bytes) to represent an extended precision floating-point number instead of using one doubleword as is done in long-form representation. Fourteen hexadecimal digits, or up to 17 decimal digits, of precision are provided by the long floating-point format.

  Seven extended precision floating-point instructions are included in this feature. They provide addition, subtraction, and multiplication operations for extended precision data, using a pair of floating-point registers, and the ability to round from long to short form or from extended to long form. An extended precision divide instruction is not provided; however, a simulator for this operation is provided in OS (discussed in Section 30).

ARCHITECTURE IMPLEMENTATION ALTERATIONS

Two alterations have been made to the system action taken on a Model 135 during the execution of certain instructions common to both System/370 and System/360 models. The first involves all instructions that check the validity of operands involved in packed decimal operations. On the Model 135, an invalid sign in an operand causes the instruction to be suppressed (never executed) rather than terminated during execution as is done on System/360 models. Suppression, rather than termination, of an instruction when an invalid sign occurs ensures that the data fields involved remain unchanged. Therefore, a routine that inspects the field that has the invalid sign can be executed when a program check occurs.

For example, when an invalid sign results from packing an entirely blank field, the sign can be corrected by programming, and transaction deletion or program termination is avoided.

The second alteration concerns the recognition of a storage protection exception during the execution of an EDIT or an EDIT AND MARK

instruction. On a Model 135 a protection exception always occurs when a pattern character is fetched from a location protected for storing but remains unchanged during the edit operation. This change eliminates unpredictable system operation during editing on a Model 135. The occurrence of a protection exception for the situation described is model dependent for System/360 models.

INTERVAL TIMER

The interval timer in decimal location 80 in the fixed processor storage area is a standard feature and has a resolution of 3.3 milliseconds instead of the 16.6-ms resolution implemented for the interval timer provided for Models 25 and 30. Its maximum time period remains 15.5 hours. The higher resolution of this interval timer eliminates many of the problems encountered in accounting routine accuracy caused by task execution durations that are less than the 16.6-ms resolution interval.

TIME OF DAY CLOCK

This new clock is a binary counter of 52 bits with a cycle of approximately 143 years. It is a standard feature. The clock is updated every microsecond. Two new instructions (SET CLOCK and STORE CLOCK) are provided to set the time and to request that the current time be stored in the specified doubleword of processor storage. The time can be set only when the CPU is in supervisor state and only when the clock security switch on the system console panel is in the enable set position.

The time of day clock can be used for more accurate time stamping than the interval timer. More accurate time of day can be maintained because, during normal system operation, the clock stops only when CPU power is turned off. (Use of certain system test modes and an error in the clock also invalidate the clock time. Thus, the clock value is invalidated during the initial microprogram load procedure by the execution of microdiagnostics at that time.) The interval timer cannot be as accurate as the clock for time of day maintenance because it is not updated when the system is in the stopped state and its updating may be omitted under certain conditions of excessive system activity. The 15.5-hour cycle time of the interval timer is also a restriction. The time of day clock better answers the timing needs of teleprocessing and real-time applications.

10:15 STORAGE AND THE CONSOLE FILE

CONTROL STORAGE

A significant new storage feature of the Model 135 is reloadable control storage (RCS) for microprogram residence. The use of writable storage for control functions adds to the advantages of using a read-only storage instead of conventional circuitry.

Control storage is implemented in the same monolithic storage as is processor storage (described in Section 10:05). All control storage is contained in one basic storage module and is separate from processor storage. However, control and processor storage share error checking logic (ECC), and address and data paths to and from the CPU.

The standard control storage size is 24K bytes, which can be expanded to 36K or 48K. Expansion is required when certain optional features or feature combinations are selected. The standard 24K of control

storage provided is divided into a fixed area and a variable area. The fixed area is located in the low order portion of control storage and it contains all the microcode required by the standard Model 135 features. No substitutions can be made for features contained in this fixed area. Microcode contained in the fixed area is assigned to fixed addresses and packaged such that a minimal amount of control storage space is unused.

The variable area in the standard 24K is assigned the microcode for a particular group of optional features. The microcode for certain other options is always allocated to the first 12K control storage increment so that when one or more of these options are selected, the first increment is required. The second 12K control storage increment must be installed in addition to the first when certain combinations of block multiplexing, 1401/1440/1460 Compatibility, and the ICA are included.

The features supported in a system with 24K of control storage are shown below. The sequence in which items are listed within an area does not necessarily indicate their relative locations.

	24K of Control Storage
Fixed area (standard features)	• Standard Model 135 instruction set (binary and decimal arithmetic, storage and fetch protection) • 16 byte multiplexer UCW's • OS/DOS Compatibility
Variable area (selective and optional features)	• 3210 Model 1 or 3215 console (one or the other is required) • Integrated File Adapter • Direct Control • Selector channels • Floating-point arithmetic (only if 3210 console is used)

Installation of any of the following features or combination of features requires the presence of the first 12K control storage increment:

- 3215 console and floating-point arithmetic
- Extended precision floating point
- Additional 48 byte multiplexer UCW's
- Block multiplexing*
- Integrated Communications Adapter*
- 1401/1440/1460 Compatibility*

* Second increment may be required also

The second 12K increment is required, in addition to the first, when either of the following feature combinations is selected:

- 1401/1440/1460 Compatibility and the ICA with the Synchronous Data Adapter

- Any combination of three of the following features:
  1) Block multiplexing
  2) 1401/1440/1460 Compatibility
  3) ICA with Synchronous Data Adapter
  4) ICA with Terminal Adapter Type I
  5) ICA with Terminal Adapter Type III

If disk cartridges that contain different feature mixes are desired by an installation, they can be requested from IBM via a special order.

As implemented in the Model 135, use of RCS instead of ROS results in system cost savings and provides improved serviceability and additional system functions:

- System cost savings result from the fact that the total amount of control storage required is reduced because control storage need not be large enough to contain all the possible microcode for the system. Since control storage is reloadable, the microcode required to support the optional features of a specific system configuration can be more efficiently packaged in the variable control storage areas when the given system microcode is customized. In addition, all microcode for a system need not be resident at all times. For example, different versions of microcode for a given system containing different features can be loaded when required, and diagnostics overlay normal system microcode when they are needed. Furthermore, the fact that a single type of writable storage can be and is used for both control and processor storage helps achieve the price performance goal of the Model 135.

- Serviceability is enhanced because of the speed and ease of engineering change installation--the new microcode need only be loaded into RCS--and because more extensive diagnostics can be provided without the necessity of adding additional control storage (all control storage is available to be used for diagnostic residence). The design simplicity of using a single type of storage in the system (one storage-addressing design, a single set of sensing circuits, a common data flow design, etc.) also benefits serviceability and reduces cost.

- Functional capability is extended by the ability to more easily support different architectures and features in one system. IBM-supplied 1400 and DOS emulator microcode and special features are quickly and easily loaded.

PROCESSOR (MAIN) STORAGE

Processor storage has a read cycle of 770 nanoseconds and a write cycle of 935 nanoseconds for CPU access to two bytes. These cycle times include the time required to fetch the next microinstruction because storage operation and microinstruction fetch are overlapped. All processor storage is contained within the CPU frame and model changes are field installable. Processor storage is available in the following increments:

Model	Storage Size (K=1024 bytes)
FE	96K
GD	144K
GF	192K
DH	240K

Error checking and correction (ECC) hardware provides automatic detection and correction of all single-bit processor and control storage errors and detection, but not correction, of all double-bit and some multiple-bit errors. ECC logic is packaged in a module separately from control and processor storage and is shared by both. The ECC feature is discussed fully in the RAS section.

The Model 135 supports a byte boundary alignment facility for processor storage. The presence of the byte-oriented operand function allows the processor storage operands of unprivileged instructions (RX and RS formats) to appear on any byte boundary without causing a specification program interrupt. Without this facility, operands must be aligned on integral boundaries, that is, on storage addresses that are integral multiples of operand lengths. Byte orientation is standard and does not apply to alignment of instructions or channel command words (CCW's).

Use of byte alignment in a program degrades instruction execution performance. However, byte orientation can be used effectively in commercial processing to eliminate the padding bytes added within records or to blocked records to ensure binary and floating-point field alignment. The smaller physical record that results from the elimination of padding bytes requires less external storage and increases effective I/O data rates. I/O-bound commerical programs, in which throughput is in almost direct proportion to the I/O data rate, can achieve performance improvement by using byte alignment for binary and floating-point data.

A program written to use byte boundary alignment will not necessarily run on a System/360 model that does not have the feature. Therefore, programs that are to run on both the Model 135 and on System/360 models without byte orientation should be written to adhere to integral boundary rules.

THE CONSOLE FILE

Control storage is loaded directly from a small read-only disk device, called the console file, which is a basic component of the Model 135. This file is located in the upper left-hand corner of the system control panel, and it reads removable prerecorded disk cartridges. A disk cartridge has an effective capacity of approximately 75,000 bytes, and a data track is divided into eight sectors for recording purposes. Data is read from the disk cartridge by the file at a rate of 33,300 bits per second.

The disk cartridge is contained in a protective eight-inch-square casing, as shown in Figure 10.15.1. When mounted on the console file, the cartridge rotates inside its casing, and data is read through a hole in the casing that exposes the data recording area. Reading from the console file is initiated by console switches and buttons. Once the file has been started, its operation is controlled by command bytes that are interspersed within the data (microcode or diagnostics) contained on the disk cartridge tracks. There are no I/O instructions

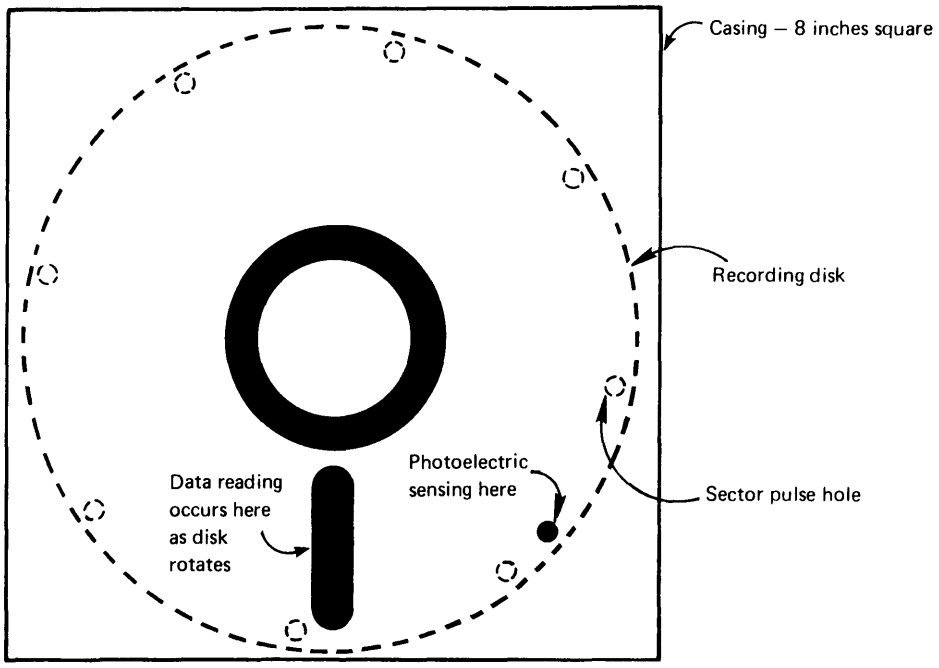


Figure 10.15.1. Disk cartridge for the console file. Shaded areas represent recording disk showing through cutout areas in casing. Dotted lines show recording disk inside the casing. Eight holes in the periphery of the disk generate, via photoelectric sensing, sector pulses to indicate beginning of sector.

or commands that a program can execute to cause reading from the file and there is no way for any installation to write data on a disk cartridge.

Prewritten disk cartridges containing all the microcode required for the specific Model 135 configuration will be shipped to each installation. One standard and one backup cartridge containing the customized system microcode will be provided in addition to other disks that contain system diagnostics. The system microcode cartridge also contains a set of system microdiagnostics, which are executed during the initial microprogram load procedure, before system microcode is loaded, to verify that the system is operating correctly. A single cartridge has enough capacity to contain the system microdiagnostics and all the available system microcode.

In order to load control storage with the system microcode, the proper disk cartridge, called the initial microprogram (IMP) disk, must be mounted on the console file, the diagnostic/file control switch on the system console panel must be in the process position, and the CPU must be in process mode. When the system is in this status, an initial microprogram load (IMPL) is initiated automatically when system power is turned on.

When power is already on, IMPL is initiated via the start console file key. Pressing this key causes power to be turned on in the file (it is normally off), and as soon as the disk is up to speed--about five seconds--reading from the console file begins. The key remains red from the time it is pressed until reading begins and then turns

white to indicate that loading has begun successfully. The key remains red if the disk does not turn, a disk is not mounted, etc.

Approximately one minute is required for execution of the system microdiagnostics and loading of control storage. During microcode loading, considerable error checking occurs to ensure that the microcode is read and loaded into control storage correctly. If an error is detected, the console file key turns red to inform the operator. The white indication and file power are turned off at the successful completion of control storage loading. The system reset microcode (just loaded) is executed, and the CPU is placed in the stopped state, ready for an IPL of the operating system.

A procedure is established by which the user or customer engineer can load temporary patch engineering changes (EC's) from an I/O device at the completion of IMPL. When temporary patches are received, the customer engineer plugs a jumper card in the system to cause a request for patch loading to be typed on the console at the completion of each IMPL procedure. Checking is performed to ensure that EC's are correctly loaded. Once microprogram patches have been made, no additional changes can be made until control storage is again loaded. EC's distributed as temporary patches will be included in the next customized system microcode disk cartridge sent to each Model 135 installation, at which time the customer engineer will remove the jumper card.

Note that when system power is turned off, the data in both processor and control storage is lost, so an IMPL must be performed when power is turned on again (the IMPL can be performed automatically, as discussed above). Normally, the cartridge containing the system microcode will stay mounted on the console file and cartridge changing will occur only when diagnostics are performed.

If necessary, a disk cartridge containing customized system microcode for a given Model 135 can be used to load control storage of another Model 135 at the same EC level that has the same control and processor storage size and the same hardware features. However, this procedure is not recommended for the following reason. The system serial number for a Model 135 is contained on the customized microcode disk cartridge sent for the system, rather than in the CPU itself. Therefore, when the STORE CPU ID instruction is issued, the serial number presented is the one loaded into control storage from the microcode disk used. Thus, when the IMP disk for a given Model 135 is used on another Model 135, the serial number recorded in any error records placed in the error recording file (SYSREC for DOS, SYS1.LOGREC for OS) during system operation will be that of the original system, not the system on which the error actually occurred.

The console file is also used for loading and executing diagnostic routines and it is a basic debugging tool for the system. A comprehensive set of fault-locating diagnostic routines is supplied to each installation on disk cartridges that can be loaded directly from the console file into the Model 135 and executed. These diagnostic routines are discussed in Section 50:15.

10:20 CHANNELS AND I/O ADAPTERS

The Model 135 user has a broad range of channel facilities from which to choose when configuring the input/output of a system. While channel functions compatible with those available on Models 25 and 30 are provided, the Model 135 offers new facilities, integrated adapters, more varied configuration capabilities, and faster channel data rates. In addition, faster I/O devices can be attached. These capabilities enable the Model 135 user to tailor a system to his I/O processing

needs, on an improved price performance basis, to increase channel throughput.

A byte multiplexer channel with 16 subchannels is standard. Optional features include additional byte multiplexer subchannels, an Integrated File Adapter for direct attachment of 2314-type direct access storage devices, an Integrated Communications Adapter for direct attachment of communications lines, one or two selector channels, and block multiplexer mode of selector channel operation.

The byte multiplexer channel is addressed as channel 0. The Integrated Communications Adapter (ICA) logically attaches to the byte multiplexer channel and thus its communication lines are assigned channel 0 addresses. The Integrated File Adapter (IFA) is always addressed as channel 1. The two selectors are addressed as channels 1 and 2 when the IFA is not installed and as channels 2 and 3 when the IFA is present.

Model 135 channels are integrated as they are on Models 25 and 30. They share with the CPU the use of control storage, use of the CPU and processor storage data flow, and use of the CPU arithmetic logic unit. All channel activity, including both data servicing and status handling, generates CPU interference. In addition, the IFA and selector channels can cause mutual interference and can interfere with the byte multiplexer channel.

All byte multiplexer operations are microprogram-controlled. IFA and selector channel operations are controlled by a combination of hardware and microprogramming. Hardware control is used for data transfer sequences to speed up chaining operations. That is, a hardware-controlled operation, called cycle stealing, is used to transfer data between processor storage and the IFA or a channel. During a cycle steal operation, the associated data count and storage address for the I/O operation are modified. A cycle steal requires 660 nanoseconds during which one or two bytes of data are transferred to or from processor storage. Microprogram activity is suspended for the duration of a cycle steal or a succession of cycle steals, after which activity is resumed. Cycle stealing is transparent to a microprogram.

Data transfer requests for the byte multiplexer channel (and the ICA) are microprogram-controlled and have the lowest servicing priority. Cycle steal requests from the IFA and the two selector channels are handled in such a way as to give a channel or the IFA the use of at least every third cycle steal. When cycle steal requests are sampled, all outstanding requests are arranged in high to low priority sequence: IFA requests, first channel requests, second channel requests. Requests are then handled in priority order, on a rotating basis (IFA, first channel, second channel, IFA, etc.) until all have been serviced before cycle steal requests are again sampled.

Comprehensive error checking has been incorporated in the basic design of the channel hardware. Checking is performed on the control logic in most areas, and standard parity checking is done on the data flow. Improved error recovery hardware has also been included (discussed fully in the RAS section).

The standard instruction set also includes a new I/O instruction, called HALT DEVICE. This instruction is specifically designed to stop an I/O operation on a particular device on a byte or block multiplexer channel without interfering with other I/O operations in progress on the channel. HALT DEVICE should always be used, instead of HALT I/O, to stop an I/O operation on a multiplexer channel.

A Model 135 channel can be interconnected to another System/360 or System/370 channel that has the optional Channel-to-Channel Adapter feature installed. The adapter itself cannot be installed on the Model 135.

BYTE MULTIPLEXER CHANNEL

The standard byte multiplexer channel provided for the Model 135 is functionally identical to the one available on System/360 Models 30, 40, and 50. However, on the Model 135 it is intended to operate primarily in byte interleaved mode, to permit several slow-speed devices to operate concurrently. While the byte multiplexer channel on the Model 135 can operate in burst mode, most unbuffered burst mode devices will overrun if placed on this channel. Data is transferred between the byte multiplexer channel and processor storage one byte at a time.

The byte multiplexer channel can have 16 or 64 subchannels. The number of subchannels in a Model 135 system is not related to the size of processor storage. The number of subchannels desired must be specified by the user when the Model 135 is ordered so that control storage for unit control words (UCW's) is allocated and initialized by the customized installation microcode disk cartridge.

The maximum number of I/O devices that can be attached to the byte multiplexer channel is a function of the number of subchannels present and the number of integrated adapters with a channel address of 0 in the system. The console printer-keyboard requires one byte multiplexer subchannel as does each communication line attached via the Integrated Communications Adapter.

The number of subchannels present determines the maximum number of concurrent I/O operations that can execute on the byte multiplexer channel. Each subchannel is associated with a unit control word. UCW's are used to store channel register data between data transfers to and from processor storage when devices are operating on the byte multiplexer channel. Byte multiplexer UCW's are contained in control storage and are 16 bytes in length.

INTEGRATED FILE ADAPTER

One Integrated File Adapter (IFA) can be installed on a Model 135. This optional feature allows direct attachment to the Model 135 of three, four, or five 2314-type drives without the necessity of installing a selector channel and disk control unit. It provides the Model 135 with a lower cost method of attaching 2314 disk storage. In addition, power supply and space requirements are reduced.

The 2319 Disk Storage Model A1 unit, which consists of three disk drives with a total maximum capacity of 87 million bytes, must be the first unit attached to the Model 135 via the IFA. The disk drives in the 2319 are functionally compatible and program compatible with 2314A-type disk storage drives, and the interchangeable 2316 Disk Pack, with a maximum capacity of 29 million bytes, is used as the storage medium. One or two additional 2314A drives, either 2312 (one drive) or 2318 (two drives) disk storage, can be attached via the IFA.

Note that 2314 disk facilities can be channel-attached to the Model 135 via 2314-A1 and 2314-B1 control units whether or not the IFA is installed. A facility with up to nine drives (one of which is a spare) consisting of 2312, 2313, and 2318 disk storage can be channel-attached using the 2314-A1 control unit. A facility with three, six, or nine drives (one of which is a spare) consisting of 2319-B1 and 2319-B2 disk storage can be channel-attached using the 2314-B1 control unit.

When 2314-type disk storage is attached to the Model 135 via the ICA, additional disk storage can be attached to only one of the two selector channels available.

The Integrated File Adapter is always addressed as channel 1. The IFA performs the functions of a selector channel and disk control unit for the disk drives attached to it and is programmed as though channel 1 and a control unit were present. Cycle stealing is used to transfer one byte of data at a time between processor storage and the IFA. The Record Overflow and File Scan features are standard on the IFA. However, 2844 Auxiliary Storage Control, the Channel-to-Channel Adapter, block multiplexing, and the Two-Channel Switch feature do not apply to the IFA.

INTEGRATED COMMUNICATIONS ADAPTER

The Integrated Communications Adapter (ICA) optional feature, together with lower cost-per-byte direct access storage (offered by the 2319) and lower cost processor storage on the Model 135, offers intermediate system users economic and flexible entry into communications-based applications. The ICA provides communication facilities compatible with those of Models 25 and 30 and thus also provides simplified transition from these models to the Model 135.

The ICA permits direct attachment to the Model 135 of up to eight low- and medium-speed communication lines. Only one ICA can be installed on a Model 135. Lines connected via the ICA are addressed and logically operate as if attached to the byte multiplexer channel via a 2701 Data Adapter Unit. One byte multiplexer UCW is required for each communication line. The eight line addresses used must be 01 through 08 (hexadecimal), and they must be assigned consecutively to the lines installed, beginning with address 01. The Model 135 can have communication lines and terminals connected via the byte multiplexer channel and 2701, 2702, and 2703 transmission control units in addition to, or instead of, via the ICA.

Unlike the 2701, which is a stand-alone hardware-controlled unit that can handle up to four communication lines, the Model 135 ICA is contained in the CPU and is both microprogram and hardware controlled. The amount of control storage required to support the ICA varies, depending upon the number of lines, the types of terminal adapters, and the features to be handled. The system microcode is customized based on user specification of the ICA configuration when the system is ordered.

The ICA supports half-duplex communication lines. They can be private, leased, or switched public telephone lines. Private or leased lines can be multidrop and the Model 135 can be the control or a tributary station. Start/stop terminals, alphanumeric display units, and binary synchronous communications are supported. Each line connected to the ICA must have attached only one of the three types of terminal adapters available. Any mixture of adapter types is permitted on the ICA communication lines. Each line also requires a modem external to the Model 135 CPU. Processor-clocked and modem-clocked (self-clocking) modems can be used and data rates up to 4800 bits per second can be handled. (See IBM System/370 Model 135 Configurator, for configuration details about modems, lines, data rates, etc.)

The following types of terminal adapter features, compatible with the same adapters for the 2701, and communication terminals are supported by the Model 135 ICA:

- Terminal Adapter Type I Model II
  1050 Data Communication System
  2740 Communication Terminal (Models 1 and 2) and 2760 Optical Image Unit
  2741 Communication Terminal
  System/7

- Terminal Adapter Type III
  2845 Display Control and 2265 Display Stations
  2848 Display Control (Models 1, 2, and 3) and 2260 Display Stations (Models 1 and 2)

- Synchronous Data Adapter Type II
  Another System/360 or System/370 with a 2701 Data Adapter Unit or 2703 Transmission Control
  Another System/370 Model 135 with the ICA and Synchronous Data Adapter Type II feature
  System/3 (Model 6 and Model 10) with the Binary Synchronous Communications Adapter
  Model 20 with the Binary Synchronous Communications Adapter
  Model 25 with the ICA and Synchronous Data Adapter
  1130 System with the Synchronous Communications Adapter
  1800 System with the Communications Adapter
  2770 Data Communication System
  2780 Data Transmission Terminal
  2790 Data Communication System (via 2715-2)
  3735 Programmable Buffered Terminal

In addition to the above, the 2701 supports 1030, 1060, and 1070 communication systems, AT&T 83B2/83B3 terminals, WU 115A terminals, and TWX-33/35* terminals. While these systems and terminals cannot be attached to the Model 135 via the ICA, they can be attached via the byte multiplexer channel and a 2701, 2702, or 2703 transmission control unit. (The Model 25 ICA supports the same terminals as the 2701 except for the 2260 and 2265 Display Stations and is functionally compatible with the 2701 when DOS BTAM is used.)

The terminal adapter features for the Model 135 ICA are functionally equivalent to their 2701 counterparts, as far as support provided is concerned, with a few exceptions as outlined below. Variations in actual implementation between Model 135 ICA and 2701 equivalent adapters are handled by the telecommunications access methods and these are transparent to the user when these access methods are used. The Model 25 ICA is compatible with the Model 135 ICA for the same functions for which the Model 25 ICA is compatible with the 2701.

See System/370 Model 135 Functional Characteristics for specifics about the CPU interference caused by operations on the ICA.

ICA Feature Equivalent to 2701 Terminal Adapter Type I Model II

This basic adapter feature is used for communication with a 1050 system, a 2740 terminal, a 2741 terminal without the Interrupt feature, or a System/7 and uses a processor-clocked modem. A data rate of up to 600 bits per second is supported.

Write Interrupt is an optional feature for this adapter that allows a terminal operator to interrupt a transmission from the Model 135. This feature, not available on the 2701, is compatible with the 2741 Break feature on the 2702 and 2703 that supports a 2741 with the Interrupt feature. Read Interrupt, not available on the 2701 or the

* Terminals which are equivalent to those explicitly supported may also function satisfactorily. The customer is responsible for establishing equivalency. IBM assumes no responsibility for the impact that any changes to the IBM-supplied products or programs may have on such terminals.

Model 25 ICA, is another feature for this adapter. It permits an executing program in the Model 135 to interrupt a transmission from a terminal as does the Type I Terminal Interrupt feature on the 2702 and 2703.

ICA Feature Equivalent to 2701 Terminal Adapter Type III

This feature is used for communication with remote 2848 Display Control and 2260 Display Stations or remote 2845 Display Control and 2265 Display Stations. Either a processor-clocked modem with a data rate of 1200 bits per second or a self-clocking modem with a data rate of 2400 bits per second can be used.

ICA Feature Equivalent to 2701 Synchronous Data Adapter Type II

This feature is used for communicating with any terminal or IBM computer that conforms to the Binary Synchronous Communication standard defined in General Information - Binary Synchronous Communications (GA27-3004). Data rates of 600 or 1200 bits per second with a processor-clocked modem and any data rate up to 4800 bits per second with a self-clocking modem are supported. (The 2701 supports up to 230,400 bps and the Model 25 ICA handles up to 4800 bps.)

The communication network can be point-to-point or centralized multipoint. There is no provision for noncentralized multipoint operations (supported by the 2701) and the Model 135 ICA rejects a SEARCH command. In a centralized multipoint operation, each line can be either a control station or a tributary station. The control station feature is standard and if a line is to have the tributary feature instead, this must be specified when the ICA is ordered.

A line that is flagged as a control station operates in the same way as a 2701 Synchronous Data Adapter without the associated Station Selection facility. A line that is flagged as a tributary station operates in the same way as a 2701 Synchronous Data Adapter with the associated Station Selection facility, and will reject a POLL command.

The Autoanswer feature is available for the Model 135 ICA, but the Autocall feature (DIAL command), available for the 2701 and the Model 25 ICA, is not.

Each binary synchronous line connected to the ICA can use any one of three data code options: EBCDIC, ASCII, or Six-Bit Transcode. Optionally, a line can have the Dual Code feature and/or the Transparent Mode feature (features also available on the 2701 and the Model 25 ICA). The Dual Code feature permits one of the three possible data code options to be used as an alternative code for the line. The line uses the primary code assigned unless the alternate code is made effective by issuing a SET MODE. The Transparent Mode feature provides for the transmission of all possible bit patterns within the selected code level. Binary synchronous lines attached via the ICA can have data code features different from one another, if desired.

SELECTOR CHANNELS

Optionally, one or two selector channels can be installed on the Model 135 in addition to the optional Integrated File Adapter. Selectors are addressed as channels 1 and 2 when the IFA is not installed and as channels 2 and 3 when the IFA is present. Data is

transferred between processor storage and the four-byte buffer in each channel, one or two bytes at a time, using cycle stealing. Single-byte transfers occur when the first or last byte of a data block is on an odd storage address, for example.

Selector channels on a Model 135 are functionally equivalent to System/360 selector channels but support significantly higher data rates than Model 30 channels. The maximum data rate that can be sustained by the first and the second selector channel, respectively, is 1.3 MB and 1.2 MB. When both selectors are operating, a maximum aggregate selector channel rate of 2.4 MB can be sustained (1.2 MB on both, 1.3 MB on the first selector and 1.1 MB on the second selector, etc.) concurrently with a data transfer rate of 312 KB on the IFA. However, CPU and byte multiplexer channel operations are locked out when this maximum channel activity occurs.

A Model 135 I/O device configuration can include direct access storage on both selector channels if the IFA is not installed. When the IFA is present, direct access storage can be placed on only one of the two possible channels and the channel selected must have the higher priority for command chaining operations.

The normal priority for command chaining channel operations is (high to low) first channel (channel 2), IFA, second channel (channel 3). The no-charge Channel Priority feature, if installed, reverses the priority of the two channels for command chaining operations, which results in a high to low priority sequence of channel 3, IFA, channel 2. The Channel Priority feature is required only if the IFA is present and direct access devices are to be placed on channel 3 instead of on channel 2. Installation of this feature affects the priority for command chaining operations only. Other priorities, such as that for cycle stealing requests, remain the same.

To attach 3330 facilities to both channels, an estimate of the probability of data overrun should be determined. Advice should be sought from IBM personnel when the application and data environment are known.

The following summarizes the I/O device configurations possible on the Model 135. The first channel has the highest priority for command chaining operations. If the Channel Priority feature is installed, the I/O devices indicated for the first channel in configurations 1 and 3 can be reversed with those indicated for the second channel.

	IFA	First Channel	Second Channel
1.	2319	2314-type (and any other Model 135 attachable device, including 2314-type, excluding 3330)	Any other Model 135 attachable device except a direct access device
2.	-	2314-type (and any other Model 135 attachable device, including 2314-type, excluding 3330)	Same as first channel
3.	2319	3330 (and any other Model 135 attachable device, including 2314-type and 3330)	Any Model 135 attachable device with a data rate not in excess of 90 KB

	IFA	First Channel	Second Channel
4.	-	3330 (and any other Model 135 attachable device, including 2314-type and 3330)	Any Model 135 attachable device including 2314 and 3330. If 3330, advice must be obtained from IBM personnel.

BLOCK MULTIPLEXER CHANNELS

Block multiplexing mode on the Model 135 is an optional, no-charge feature. It permits all installed selector channels to operate as block multiplexer channels. Installation of block multiplexer mode is required when 3330's are attached to the Model 135 and OS is used. The feature is not required when DOS is used because DOS does not support block multiplexer mode.

The setting of a channel mode bit in a control register determines whether a channel with block multiplexing capability operates in block multiplexer or selector mode. When the block multiplexing feature is not installed, this bit has no effect on channel mode. The mode bit is set to selector mode at IPL or on system reset and can be altered by programming at any time. When a START I/O instruction is issued to a channel with block multiplexing capability on the Model 135, the current setting of the channel mode bit determines the mode in which the subchannel involved will operate.

The block multiplexer channel is designed to increase system throughput by increasing the amount of data entering and leaving the system in a given period of time (the effective data rate). Better use of channel time can be achieved by operating the channel in block multiplexing mode. A single block multiplexer channel can support interleaved, concurrent execution of multiple high-speed channel programs. The block multiplexer channel can be shared by multiple high-speed I/O devices operating concurrently, just as the byte multiplexer can be shared by multiple low-speed devices. Like the byte multiplexer, the block multiplexer channel has multiple subchannels, each of which has an associated UCW in control storage and can support one I/O operation.

A block multiplexer channel functions differently from a selector channel in the way in which it handles command-chained channel programs. A selector channel or a block multiplexer channel operating in selector mode executing a command-chained channel program is busy during the entire time the channel program is in operation, whether data transfer is occurring or not. A block multiplexer channel executing a command-chained channel program has the ability to disconnect from the operational channel program during certain non-data transfer operations. That is, a block multiplexer channel can be freed during a nonproductive activity, for example, during disk seeking and most record positioning, thereby allowing more data to be transferred per unit of channel busy time.

Block multiplexing operates as follows. Assume a block multiplexer channel is executing a channel program consisting of multiple command-chained CCW's. When channel end is presented without concurrent device end, the channel disconnects from the channel program and becomes available for an I/O operation on another device--even though the disconnected channel program is not complete. At channel disconnect time the subchannel and the device's control unit retain the information necessary to restart the disconnected channel program.

When the device signals that it is again ready for the channel (by presenting device end), its control unit attempts to regain use of the channel. If the channel is free at this time, the channel registers are reloaded with the information previously saved (in the device's

UCW), and the disconnected channel program is resumed at the appropriate CCW. If the channel is busy when reconnection is requested, the device must wait until it becomes available. Once multiple channel programs have been initiated on one channel, the interleaving of data transfer operations is controlled by block multiplexer channel hardware and the control units of the devices operating in block multiplexing mode.

To facilitate channel scheduling on block multiplexer channels, a new interrupt condition, called channel available, has been defined. At disconnect time for a channel program, the block multiplexer channel is available for the resumption of an uncompleted channel program previously started, or another channel program can be initiated. A channel available interrupt occurs at disconnect time to indicate channel availability if a START I/O, TEST I/O, TEST CHANNEL, or HALT DEVICE instruction was issued previously while the block multiplexer channel was busy.

Two additional facts should be noted about block multiplexer channel operations:

1. When multiple channel programs are operating concurrently in block multiplexing mode, a device can regain control of the channel only when the channel is not busy. Thus, only cyclic devices (such as direct access devices with rotational position sensing) or buffered devices (such as the 2540 Card Read Punch and the 1403 Printer) can disconnect during the execution of a command-chained channel program on a block multiplexer channel and resume operation later.

2. Data transfer operations for concurrently operating devices on a block multiplexer channel are interleaved on a first-come, first-served basis as the desired records become available. Thus, devices are serviced in the order in which their records become available, not necessarily in the order in which their channel programs are initiated.

Seventeen UCW's are provided for each block multiplexer channel in the Model 135. Sixteen of these UCW's are nonshared and designed to be used with control units capable of block multiplexing. A UCW (or subchannel) is referred to as nonshared if it is associated and can be used with only one device. The seventeenth UCW is shared among all the control units on the channel that are not associated with the nonshared UCW's.

A shared UCW can be used by a set of devices, one device at a time. A shared UCW generally is assigned to a control unit that has multiple devices attached, only one of which can be in operation at a time (such as a tape control unit). Devices that use the seventeenth UCW will not block multiplex (disconnect during command chained channel programs as discussed previously).

Normally, the seventeen UCW's will be used to support block multiplexing operations on one or two control units concurrently with one non-block-multiplexed operation on the same channel. In this case, the Model 135 can handle block multiplexing on a channel for either one control unit with up to 16 similar devices or two control units with up to eight similar devices on each. If more than two control units are to use the 16 nonshared UCW's, the control unit addresses used must follow a certain rule, which is discussed later.

Examples of devices that can block multiplex on the Model 135 when attached to a nonshared UCW are:

- 3330 facilities - one UCW per drive in the facility is required
- 2540 Card Read Punch units - one UCW for the reader and one for the punch
- 3505 Card Reader and 3525 Card Punch - one UCW for each reader and for each punch
- 1403 Printers attached to a 2821 Control Unit - one UCW per printer
- 3211 Printers - one UCW per printer

When attached to the Model 135, magnetic tape units and direct access devices without rotational position sensing, such as the 2311, 2321, 2303, 2314, and 2319, should be associated with the shared UCW of a block multiplexer channel, as are all devices not assigned to a nonshared UCW.

The control units that are to perform block multiplexing operations are designated when the system is installed. Assuming only two control units are to multiplex on the channel, the four-bit control unit addresses of these two control units are plugged into the block multiplexer control unit address byte in the block multiplexer channel by the customer engineer. Any permissible control unit addresses can be used for block multiplexing operations. If only one control unit on a channel is to be multiplexed, its address is placed in both halves of the address byte in that channel.

The 16 nonshared UCW's are permanently assigned to the control unit(s) whose address(es) are plugged into the system by the customer engineer. The seventeenth shared UCW must be dynamically assigned during system operation. Each time a START I/O (SIO) instruction is issued to a channel operating in block multiplexing mode, the channel compares the four-bit control unit address indicated by the SIO instruction to the two control unit addresses in the block multiplexer control unit address byte in the channel. The result of the comparison indicates whether or not the seventeenth UCW should be assigned to the I/O operation. One of the following occurs:

1. If the control unit address in the instruction does not compare equal to either control unit address in the channel, the seventeenth UCW for the channel is assigned to the indicated control unit for the duration of the operation, after which the UCW again becomes available for assignment.

2. If the control unit address in the instruction compares equal to both control unit addresses in the channel, one of the 16 nonshared subchannels is used according to which one of the 16 possible device addresses for the control unit is indicated by the device address.

3. If the control unit address in the instruction compares equal to only one of the control addresses in the channel, the channel checks the I/O device address to determine whether the address is in the range of 0 to 7. If the address is in the 0 to 7 range, one of the eight nonshared UCW's assigned to that control unit address is used according to which one of the eight possible device addresses is indicated in the I/O instruction. The channel can handle block multiplexing operations on only eight devices on a control unit when two (or more) control units on a channel are capable of multiplexing. Thus, if the device address is above 7, the seventeenth UCW is assigned for the duration of the I/O operation, after which it is again available.

As already indicated, more than two control units can make use of the 16 nonshared block multiplexer UCW's. However, since a nonshared UCW is used only when the control unit address specified in the SIO instruction compares equal to one of control unit addresses in the address byte in the channel, no more than two unique control unit addresses can be assigned to the three or more control units that are to use the nonshared UCW's. Thus, all control units that are to block multiplex using the same set of eight nonshared UCW's must have the same control unit address, and the devices attached to these control units must have unique four-bit device addresses in the range of 0 to 7.

For example, an eight-drive 3330 facility could be assigned to use one of the sets of eight nonshared UCW's. The other set of eight could be shared by a 3505 Card Reader and 3525 Card Punch (only one control unit position is required for these units) and a 2821 control unit with multiple printers attached (two in this example). The 3505 and 2821 would have the same control unit address (different from that of the 3830 control unit) while the card reader, the card punch, and each printer would have a unique device address in the range of 0 to 7. Such a configuration supports concurrent block multiplexed operation of eight disk, one reader, one punch, and two printer channel programs together with one non-block multiplexed operation on the seventeenth subchannel.

10:25 BLOCK MULTIPLEXING OPERATIONS WITH ROTATIONAL POSITION SENSING DEVICES

Rotational position sensing (RPS) and multiple requesting are standard on 3330 facilities. These two functions, together with block multiplexing, are designed to increase system throughput by increasing channel throughput. IBM-supplied support of these three features is provided only by OS.

The presence of RPS in the control unit of a direct access device enables it to operate in block multiplexing mode. The use of rotational position sensing reduces the number of channel programs that have to be initiated for direct access devices that require an arm positioning seek (such as the 3330 facility), frees channels more often during direct access device operations--specifically, during most of the time required to position a track to a desired record--and permits disk channel programs to be initiated sooner on block multiplexer channels than is possible with selector channels.

Multiple requesting is implemented in a direct access device control unit to enable it to handle concurrent execution of multiple RPS channel programs. The control unit of the 3330 facility, for example, can simultaneously control eight RPS channel programs, one on each of its drives.

In order to overlap seek operations for current direct access devices without RPS, channel scheduling routines must initiate two channel programs for each record read or write. The first is a stand-alone seek, which frees the channel as soon as the control unit accepts the seek address. (The control unit is also free during arm movement.) At the completion of the seek, a device-end interrupt is presented, and the data transfer channel program is subsequently initiated to search for the desired record and transfer the data. A selector channel is busy during the entire search operation (execution of the SEARCH command by the control unit) that locates the desired disk record on the track. Search time can be significantly greater than data transfer time for disk records smaller than half a track in size. Search time averages one-half of a rotation for a read or write (8.3 ms for a 3330) and requires a full rotation, less record write time, for a write verification chained from a write.

Use of RPS reduces the time the channel is busy during the search for a disk record. It permits the SEARCH command to be initiated just before the desired record is to come under the read/write heads, that is, when the desired rotational position is reached. To accomplish this, a "sector" concept is employed. The tracks in each cylinder of a direct access device are considered to consist of equally spaced sectors (the number of sectors varies by device). Track formatting is unchanged but each record has a sector location as well as a record address. A sector is not physically indicated on disk tracks, but is the length of the track arc that passes under the read/write heads in one sector time. For the 3330 facility, for example, sector time is defined to be approximately 130 microseconds. Thus, there are 128 sectors per track on the 3330.

A disk control unit with RPS and multiple requesting can determine the sector currently under the heads of each of its drives. A sector counter is contained in each drive. The counter is incremented once every sector time period and set to zero each time the index marker passes under the heads. The sector in which a record falls is a function of the length of all records that precede it and of its sequential position on the track. Therefore, sector location can be calculated for fixed-length records.

Two new disk commands are provided for use with rotational position sensing:

SET SECTOR
READ SECTOR

If the sector address of a record is known or can be calculated, a SET SECTOR command can be included in the disk channel program to cause the control unit to look for the designated sector. Once the control unit accepts the sector number provided by a SET SECTOR command, both the block multiplexer channel and the disk control unit disconnect and are available for another I/O operation. When SET SECTOR is used for positioning, the time the channel is busy during the search for a record is reduced from an average of 8.3 ms to an average of 250 microseconds for the 3330 facility. (Allowing for the worst case of speed variation and for disk pack interchange, the search time for a record, from sector found to beginning of desired record, can vary from 120 microseconds to 380 microseconds on a 3330 facility.)

The READ SECTOR command is useful for sequential disk processing and for write verification. When chained from a READ, WRITE, or SEARCH command, READ SECTOR provides the sector number required to access the record processed by the previous CCW. This sector number can be used to reposition the track in order to verify the record just written or in order to read or write the next sequential record. These two new sector commands, used in conjunction with the block multiplexer channel, permit a single command-chained channel program, which frees the channel and disk control unit during seek and rotational positioning operations, to be initiated for each disk operation.

When record ID is known, the two channel programs shown below illustrate direct retrieval of a record from an OS BDAM data set on a direct access device without RPS, such as the 2314 (key was not written). The seek operation can be overlapped with other seeks and one data transfer operation on the same selector channel. (Commands shown in the sample channel programs that follow are only those that illustrate the advantage of RPS. Thus, commands such as SEEK HEAD and SET FILE MASK, which are used by data management to ensure correct operation, are not shown.)

Channel program 1. Initiate the stand-alone seek to position the disk arm.

Command Chaining Flag	Command		Selector Channel and Disk Control Unit Status
	SEEK	(Seek address)	Free as soon as the control unit accepts the seek address

Channel program 2. Initiate the data transfer operation after the seek is complete.

Command Chaining Flag	Command		Selector Channel and Disk Control Unit Status
CC	SEARCH ID EQ	(ID - sequential position on the track)	Busy (for 12.5 ms on the average for a 2314)
CC	TIC	(Back to search if ID not equal)	
	READ DATA	(Processor storage address of input area)	Busy

When the sector address is known or can be calculated, the channel program below illustrates direct retrieval of a record from the same BDAM data set on a 3330 facility attached to a block multiplexer channel. The records are fixed-length standard format and sector numbers are calculated from record ID (by data management).

Channel program 1. Initiate the seek and data transfer operation.

Command Chaining Flag	Command		Block Multiplexer Channel and Disk Control Unit Status
CC	SEEK	(Seek address)	Free during arm motion
CC	SET SECTOR	(Sector number of sector preceding desired record)	Free until sector found
CC	SEARCH ID EQ	(ID - sequential position on track)	Busy (250 microseconds average for a 3330)
CC	TIC	(Back to search if ID is not equal. With the logic shown, the first ID inspected normally is that of the desired record, and the TIC command is not executed.)	Busy
	READ DATA	(Processor storage address of input area)	Busy

The preceding example indicates the advantages of rotational position sensing and block multiplexing:

- Only one channel program is required to locate a disk record and transfer the data, thereby eliminating a stand-alone-seek I/O interrupt and the I/O supervisor processing required to schedule a data transfer channel program. A channel available interrupt may occur, however, during channel program execution.
- The channel and disk control unit are free during arm motion and rotational positioning, allowing seek and set sector operations to be overlapped with other I/O operations on that control unit and channel. Implementation of multiple requesting permits a disk control unit to control concurrent execution of multiple RPS channel programs in order to overlap seek and set sector operations for its drives.

Performance improvement gains achieved on block multiplexer channels are not due entirely to the fact that direct access device rotational delays are overlapped. Also important is the ability to initiate seek commands a number of milliseconds earlier, because a block multiplexer channel is free. The initiation of stand-alone seeks on a selector channel is delayed during search and data transfer operations. On a block multiplexer channel, seeks can be initiated during rotational positioning, since the channel and disk control unit are not busy.

SUMMARY OF BLOCK MULTIPLEXING OPERATIONS WITH I/O DEVICES

The following summarizes how direct access devices without and with RPS and other I/O devices operate on a block multiplexer channel on a Model 135 when executing a command-chained channel program.

1. Direct access devices without RPS (2311, 2321, 2314, 2319) assigned to a shared subchannel (the seventeenth UCW) operate in the same way whether the channel is in block multiplexer or selector mode. That is, the channel and the disk control unit are busy during the entire time a command-chained disk channel program is in operation. Thus, there is no disconnection after a chained seek. (If a direct access device without RPS is assigned to a nonshared subchannel, then there is disconnection after a command chained seek.)

2. The 3330 facility assigned to a nonshared subchannel executing a command-chained channel program on a block multiplexer channel disconnects after the control unit accepts an arm positioning seek that causes arm movement. Reconnection is attempted when the arm reaches its destination and signals device end. The 3330 facility also disconnects when its control unit accepts a SET SECTOR command. When the sector specified arrives under the read/write heads, the control unit attempts to reconnect and resume the CCW chain. If the channel is busy, the control unit repeats the reconnection procedure each time the specified sector position is reached.

3. All tape drives operate exactly the same whether the channel is in block multiplexer or selector mode and when attached to the shared subchannel. That is, the channel is busy during the entire time a command-chained channel program is in operation.

4. Buffered card and print devices (or devices operating with buffered control units) disconnect during the mechanical motion of the device. Reconnection occurs later to fill or empty the associated buffer. For example, a 1403 Printer attached to a 2821 control unit connected to a nonshared subchannel on a channel operating in block multiplexing mode disconnects from the channel during print time and carriage motion. Reconnection occurs when the channel is free to transfer the data for the next line to the 2821 buffer in burst mode.

5. Any other I/O device that presents channel end without simultaneous device end disconnects from a block multiplexer channel when command chaining if its control unit is designed for such disconnection and it is assigned to a nonshared subchannel. Disconnection does not occur if such devices are assigned to the seventeenth shared subchannel.

10:30 SYSTEM CONTROL PANEL AND SYSTEM CONSOLE

SYSTEM CONTROL PANEL

The system control panel attached to the end of the Model 135 CPU contains all the switches and indicators required to operate the system. The console layout--its location and the grouping of controls and indicators--was designed with human factors in mind. The five-foot-high CPU frame that contains the control panel permits most operators an unrestricted view of the machine area.

A new item in the operator control section is the clock security lever switch, which is used in conjunction with programmed setting of the time of day clock. Other new buttons and switches have been added for control of the console file. They provide the capability of loading data (microcode or diagnostics) from the console file, which itself is another new feature of the system control panel.

The capability of clearing processor storage from the system control panel is also provided. If the new enable system clear pushbutton is held in when the system reset or load button is pushed, processor storage and storage protect keys are set to zero with correct parity.

SYSTEM CONSOLE

The microprogram-controlled 15-cps 3210 Model 1 Console Printer-Keyboard can be attached only to the right-hand extension of the Model 135 console reading board for use as the operator console device. Alternately, the 3215 Console Printer-Keyboard with a print speed of 85-cps can be used. The 1052 Model 7 Printer-Keyboard cannot be attached as the primary system console device.

The adapters for the 3210 and the 3215 are integrated attachments. The console printer-keyboard installed appears as if attached to the system via the byte multiplexer channel but does not use one of the eight available control unit positions on the channel; however, the console does require one of the byte multiplexer UCW's. The console address can be either 009 or 01F and must be specified when the system is ordered so that the system microcode disk reflects the correct address.

The two new printer-keyboards are functionally compatible and program compatible with each other and with the 1052. Their keyboards are the same as that of the 1052 except that the alternate coding key has been removed from the new printer-keyboards and the EOB (now called END) and cancel keys are separate pushbuttons.

Both the 3210 Model 1 and the 3215 have a new alter/display mode of operation (not implemented for the 1052), which will be of benefit to customer engineers and operators. After the system is placed in manual mode, this new mode is set by pressing the alter/display key, which places the console under microprogram control. In this mode, data can be placed in, or printed from, the following:

- Processor storage
- General, floating-point, and control registers
- The current PSW
- A storage protect key
- Control storage (display only)

A mnemonic is entered to indicate the function to be performed. Other data, such as the starting storage address and the data to be entered, must be supplied in hexadecimal format. Both uppercase and lowercase may be used. If an error is made (incorrect mnemonic, storage address, or hexadecimal data characters, etc.), the microprogram detects the error and the operation can then be restarted. The microprogram also handles carriage returns automatically during data displays and prints eight words of hexadecimal characters per line until the end or alter/display key is depressed.

10:35 STANDARD AND OPTIONAL SYSTEM FEATURES

STANDARD FEATURES

Standard features for the System/370 Model 135 are:

- Instruction set that includes binary and decimal arithmetic, the new general purpose instructions, and the instructions required to handle the new architecture. New standard instructions for the System/370 Model 135 are:

COMPARE LOGICAL CHARACTERS UNDER MASK
COMPARE LOGICAL LONG
*HALT DEVICE
INSERT CHARACTERS UNDER MASK
*LOAD CONTROL
MOVE LONG
*SET CLOCK
SHIFT AND ROUND DECIMAL
*STORE CHANNEL ID
STORE CHARACTERS UNDER MASK
STORE CLOCK
*STORE CPU ID
*STORE CONTROL
(START I/O FAST RELEASE, a new System/370 privileged instruction functionally implemented on the Model 165, is executed as a START I/O on the Model 135.)

- Instruction retry
- Interval timer
- Time of day clock
- Expanded machine check interrupt
- Reloadable control storage - 24K
- ECC on processor and control storage
- Byte boundary alignment
- Storage and fetch protection
- Byte multiplexer channel (with 16 subchannels)
- Channel retry data in a limited channel logout area (ECSW)
- OS/DOS Compatibility
- Console file for microcode and diagnostic routine loading

*Privileged instructions

OPTIONAL FEATURES

Optional features for the System/370 Model 135, all of which can be field installed, are:

- Floating-point arithmetic (no-charge feature)
- Extended precision floating point. (Floating-point arithmetic is a prerequisite.)
- 1401/1440/1460 Compatibility (no-charge feature)
- Control storage expansion to 36K or 48K
- Direct Control (includes External Interrupt feature)
- Additional 48 byte multiplexer subchannels (no-charge feature)
- Integrated File Adapter and 2319 Disk Storage (channel 1 position only)
- Integrated Communications Adapter
- One or two selector channels
- Block multiplexer mode for all installed selector channels (no-charge feature)
- 3210 Model 1 Adapter** - for attachment of a 3210 Model 1 console
- 3215 Adapter** - for attachment of a 3215 console

** Either the 3210 Model 1 or the 3215 Console Printer-Keyboard must be installed as the primary operating system console

Note: The 3046 Power Unit is required in all Model 135 configurations.

20:05 I/O DEVICE SUPPORT

Most presently announced I/O devices and telecommunications terminals that can be attached to System/360 Models 25 and 30 can be attached to the System/370 Model 135. The following announced I/O devices and features for System/360 and/or System/370 are not included in Model 135 configurations:

1052-7	Printer-Keyboard
1231	Optical Mark Page Reader
1285	Optical Reader
1404	Printer
1412	Magnetic Character Reader
1418	Optical Character Reader
1428	Alphameric Optical Reader
1445	Printer
1827	Data Control Unit (for attachment of 1800 system analog and/or digital control units to the Model 135)
2150	Console
2301	Drum Storage
2302	Disk Storage
2305	Fixed Head Storage Facility
2319-A2	Disk Storage
2560	Multi-Function Card Machine (Used on the Model 25 with Model 20 emulation)
3210-2	Console Printer-Keyboard
7340	Hypertape Drive
7772	Audio Response Unit

Selective Tape List feature on the 1403 Printer

The 1287 Optical Reader and 1288 Optical Page Reader can be attached to a byte multiplexer channel only. In addition, 2361 Core Storage cannot be attached to a Model 135.

I/O devices for the Model 135 announced with System/370 models are:

- The 3330 facility - attaches to a selector or block multiplexer channel
- The 3211 Printer - attaches to any Model 135 channel
- The 3803/3420 Magnetic Tape Subsystem - should be attached only to a selector or block multiplexer channel on the Model 135
- The 3505 Card Reader and the 3525 Card Punch - attach to any Model 135 channel

The 3330 facility represents significant advancements in direct access device technology. It represents the latest direct access device with removable, interchangeable disk packs and embodies new data recording and access technology. The 3330 provides larger online data capacity, faster data rates and access, and expanded error correction features. Rotational position sensing and multiple requesting are standard features.

The 3211 Printer offers almost twice the print speed of the 1403-N1 and new features designed to reduce operator intervention.

The 3803/3420 tape subsystem incorporates all the latest advances in tape speed, design, and technology and offers new features and enhanced reliability, availability, and serviceability to 2400-series magnetic tape unit users.

The 3505 and 3525 80-column card units, embodying many new design features to enhance reliability, offer a variety of speeds and new functions and many new operator-oriented facilities.

The major new characteristics of these devices are discussed in the following subsections.

20:10 3330 DISK STORAGE AND 3830 STORAGE CONTROL

The 3330 facility is a modular, large-capacity, high-performance direct access storage subsystem. The 3330 facility consists of 3830 Storage Control and from one to four 3330 Disk Storage modules. A 3330 module contains a pair of independent disk storage drives, as shown in Figure 20.10.1. The new removable 3336 Disk Pack is used for data storage. Usage meters are contained in the 3830 control unit and in each 3330 module.

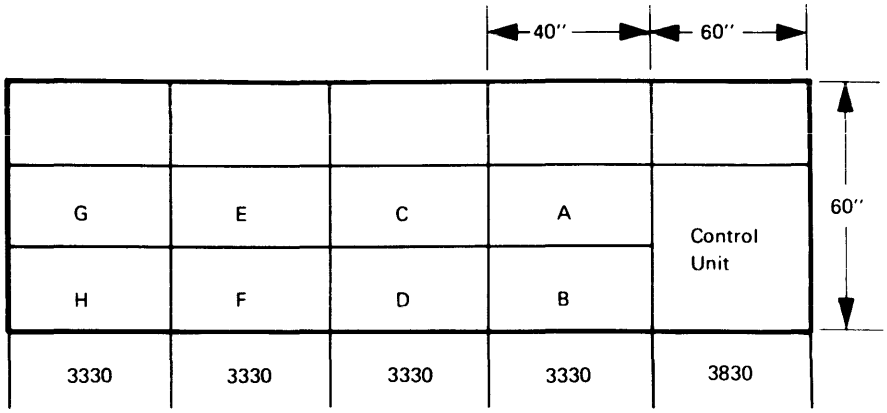


Figure 20.10.1. The 3330 facility

Drives are mounted in powered drawers that are opened and closed by a switch on the operator control panel on the 3330 module. Logical address plugs are supplied, as for the 2314, in addition to a CE service plug. The latter is used when inline diagnostics are to be executed.

Facility configurations and maximum capacities, using full-track records, are shown below.

3830 Storage Control + one 3330 module	200 megabytes
3830 Storage Control + two 3330 modules	400 megabytes
3830 Storage Control + three 3330 modules	600 megabytes
3830 Storage Control + four 3330 modules	800 megabytes

Functionally, the 3330 facility provides more capabilities than the 2314, particularly in the areas of performance and availability. The 3330 supports all the standard 2314 commands (except the file scan commands) in addition to several new operations, including RPS and error recovery commands. (Table 20.10.3, at the end of this subsection, compares 3330 and 2314 features.) The 3330 facility also is an attractive growth device for the 2321 Data Cell Drive.

The new, removable 3336 Disk Packs are interchangeable across 3330 disk drives but are not interchangeable with the 2316 Disk Packs used on 2314 disk drives (Table 20.10.2 compares disk pack characteristics). Like 2316 packs, 3336 Disk Packs will be initialized in the factory with home addresses and capacity records (R0). Up to 20 defective tracks per pack will be flagged and have alternates assigned. The

quick DASDI routine (part of the IEHDADSR utility), currently available for processing 1316 and 2316 packs, will support 3336 packs. Quick DASDI writes the volume label, the VTOC, and IPL records, if requested, but bypasses track analysis. It also determines the number of flagged tracks and places this data in the VTOC.

Table 20.10.1 compares the capacity and timing characteristics of the 3330 facility with those of the 2314 facility and the 2321 Data Cell Drive. The increase in capacity achieved by replacing a 2314 or a 2321 with a 3330 depends upon the block size chosen for the data on the 3330. For example, if the 2314 full-track block size of 7294 bytes is maintained for a given data set on the 3330 to avoid programming changes, the 3330 yields a 91% increase in full-pack capacity (almost twice the capacity). However, reblocking to a full track on the 3330, 13,030 bytes, yields a 242% full-pack-capacity increase. If there is not enough processor storage available to allocate I/O areas of 13,030 bytes, lowering the 3330 block size to one-half of a 3330 track yields a 239% increase in full-pack capacity.

Table 20.10.1. Capacity and timing characteristics of the 3330 and 2314 facilities and the 2321 Data Cell Drive

Characteristic	3330	2314 (A series)	2321
Capacity in bytes truncated to the nearest thousand (full-track records)			
Pack or cell	100,018,000	29,176,000	39,200,000
Facility or data cell drive			
2 drives/cells	200,036,000	58,352,000	78,400,000
4 drives/cells	400,073,000	116,704,000	156,800,000
6 drives/cells	600,109,000	175,056,000	235,200,000
8 drives/cells	800,146,000	233,408,000	313,600,000
10 cells	-	-	392,000,000
Access time in ms			
Maximum	55	130	600 (for strip select and load)
Average	30	60	175 (minimum for strip select and load)
Average cylinder-to-cylinder	10	25	95 (on a strip)
Rotation time in ms	16.7	25	50 (strip on drum)
Rotation speed (rpm)	3600	2400	1200
Data transfer rate (KB)	806	312	55

Table 20.10.2. 3336 and 2316 Disk Pack characteristics

Characteristic	3336	2316
Number of disks per pack	12	13
Number of recording disks	10	11
Number of recording surfaces (recorded tracks per pack)	19	20
Disk thickness in inches	.075	.050
Disk diameter in inches	14	14
Disk pack weight in pounds	20	15
Disk pack maximum capacity in millions of bytes	100	29.1
Full-track capacity in bytes	13,030	7294
Cylinders per pack	404 plus 7 alternates	200 plus 3 alternates
Tracks per cylinder	19	20
Tracks per pack	7676	4000

If a 2321 is replaced by a 3330, six full-track blocks of data from the 2321 (2000 bytes/2321 track) can be placed on each 3330 track, if full-track blocking is used, for a total of 92,112,000 bytes per 3336 pack (12,000 bytes times 7676 tracks per 3336). Thus, slightly over four 3336 packs provide the capacity equivalent of ten data cells, or a full 2321 drive, if full tracks are used. Ten full data cells, blocked full track, also can be contained in slightly more than four 3336 packs if half-track blocking is used on the 3336.

Self-formatting records are written on 3336 packs in the same manner as they are on 2316 packs. However, each physical area written (count, key, and data) has a field of error correction code appended to it for the purpose of data validity checking by the control unit instead of the cyclic check area used on the 2314.

The 3830 control unit is microprogram controlled. Read/write monolithic storage contained in the control unit is used for microprogram residence. The control unit also contains a device that reads interchangeable disk cartridges (identical to the console file device). This device is used for microprogram backup storage and for storage of nonresident diagnostics for the 3330 facility. During a 3330 facility power-on sequence, the functional microprogram is loaded from the device into control storage within the control unit. Therefore, many engineering changes can be installed merely by replacing the disk cartridge in use with another cartridge that contains the new microprogram.

The 3330 facility also incorporates new error detection, correction, and logging features, designed to improve its availability and serviceability. The following features are implemented in the 3330 that are not provided by previously announced direct access devices:

- I/O error routine correction of recoverable data errors on read operations with data supplied by the control unit in sense bytes
- Command retry initiated by the control unit to attempt hardware correction of certain errors without programming assistance
- Error logging by the control unit in its control storage of successful command retry operations
- Inline diagnostic tests contained on disk cartridges, which can be run on a single drive to diagnose hardware malfunctions while

other drives in the facility continue normal operations. (Inline diagnostics are provided currently only for 2314 facilities.)

Recovery of correctable data area errors. When the control unit detects a correctable data error during the reading of the data portion of a record, it generates the information necessary to correct the erroneous bytes. The sense bytes presented by the control unit contain a pattern of corrective bits and a displacement value to indicate which of the bytes transferred to processor storage contain the errors. The disk error recovery program need only EXCLUSIVE OR (logical operation) the corrective bit pattern with the error bytes in the input area in processor storage to correct the errors.

Command retry. Error correction (without programming assistance) is performed by a channel/control unit command retry procedure without an intervening I/O interrupt in the following five situations:

1. When a correctable data error occurs during a search or read operation on home address, record count, or record key.

   During a search or read operation, the home address, count, or key read from the disk track is placed in a buffer in control storage within the control unit. When a correctable data error occurs, the control unit corrects the data in the buffer and reissues the command that caused the error. During reorientation to the record, the control unit disconnects and frees the block multiplexer channel. When the failing search or read command is reexecuted, the corrected data in the buffer is used instead of the data actually on the track.

2. When an uncorrectable data error is detected on any portion of the record during a read or a search operation.

   The failing CCW is reissued twice by the control unit. If one of the two retries is successful, the channel program continues normally.

3. When a seek malfunction is detected.

   The control unit retries the command ten times in an attempt to position the arm correctly.

4. When an alternate or defective track condition is recognized before data transfer begins.

   The control unit determines the location of the alternate or defective track (from R0 on the track), initiates a seek to this track, and orients to the index point. When this sequence completes, the original command is reissued by the control unit. This is a programmed procedure for previously announced System/360 direct access devices.

5. When a command overrun (or late command-chaining) condition occurs during execution of a channel program because of interference from another channel or the CPU.

   The control unit initiates a retry of the command that was late.

Error logging. Usage and error counters for each drive in the facility are maintained continuously in the control unit. The usage counters are used to accumulate the number of bytes read and seeks issued. The error counters are used to accumulate the number of seek, correctable data, and uncorrectable data errors that were retried successfully by a command retry procedure, as already described. When a counter reaches its threshold or when a pack is removed from a drive,

the control unit indicates the condition via a unit check when the next I/O operation is initiated to the drive. Counter data can be obtained and counters can be reset by issuing a SENSE or READ LOG command. These statistics can then be logged in the system error data set for later diagnosis.

Inline diagnostic tests. The 3830 control unit can execute diagnostic tests on a malfunctioning drive while normal operations take place on the remaining drives in the facility. When the CE inserts the service address plug in the malfunctioning drive, diagnostic programs contained on a disk cartridge are read by the device in the control unit. Diagnostics can be executed on that drive by the customer engineer using the CE panel on the 3830 control unit. Operationally, the drive is offline to the control unit, and physically the drive is offline to the operating system.

Online testing of the 3330 facility can be performed under OLTEP control, as usual. Both OLT's and diagnostic programs contained in the OLT library can be executed on a malfunctioning drive via OLTEP. The diagnostic tests are loaded into control storage in the control unit from the OLT library. Operationally, the 3330 drive is online to the control unit but is logically offline to the operating system.

Inline and online testing allows CE diagnosis and repair of most 3330 failures without the necessity of taking the entire 3330 facility out of the system configuration.

The 3330 facility offers more than additional capacity, faster access, and attractive price performance. The 3330 facility is actually a subsystem in itself. The control unit can control the concurrent execution of one RPS channel program on each of its drives and can handle functions such as error correction and logging, which normally must be programmed, thereby relieving the control program of these activities. In addition, the availability and serviceability of the 3330 are improved by the implementation of new automatic error correction features, by use of inline diagnostics, and by the speed and ease of engineering change installation. These factors add to the improvement of total system availability.

Table 20.10.3. Hardware features of 3330 and 2314 facilities

Feature	3330	2314 (A series)
Number of drives per facility	2, 4, 6, or 8	1,2,3,4,5,6,7, or 8 (A ninth can be included as a spare only.)
Removable interchangeable disk packs	Yes	Yes
Removable address plugs	Yes	Yes
Record Overflow feature	Standard	Standard
File Scan feature	Not available	Standard
Multiple track operations	Standard	Standard
Two-Channel Switch	Optional	Optional
Second control unit (to permit two concurrent data transfer operations on a facility)	Not available	Optional (2844 Auxiliary Storage Control)
Rotational position sensing	Standard (128 sectors/track)	Not available
Multiple requesting	The control unit can concurrently handle one channel program on each of its drives.	Not available
Command retry by control unit and channel	Standard	Not implemented
Error correction data presented by control unit	Yes	No
Writable storage in control unit loaded from a disk cartridge	Yes	No
Inline diagnostic tests initiated via the CE panel in the facility	Standard	Standard
Inline diagnostic tests initiated via the system console	Standard	Not implemented

20:15 THE 3211 PRINTER

The 3211 is a high-speed line printer with front printing and new features designed to reduce operator intervention. The 3211 can print 2000 alphameric lines per minute (with a 48-character set) and is designed to be used in any installation that has a high volume of print activity. The 3211 attaches to all System/370 models and to System/360 Models 30 and up.

The 3211 has a standard 132-print-position line, which can be expanded to 150 positions as an option. The number of print positions does not affect printing speed. The Universal Character Set (UCS) feature is standard and the interchangeable train cartridge contains 432 graphics. The cartridge character arrangement is unrestricted and can be alphabetic, numeric, or special characters in any combination. When the character arrangement is optimized for specific printing loads, speeds of up to 2500 lines per minute can be attained.

The 3211 attaches to a 3811 control unit. Unlike some models of the 2821 control unit, which can handle multiple devices, a 3811 controls only one 3211 Printer. The commands used for a 1403 Printer are a subset of those provided for the 3211 Printer. Print, skip, and space commands for the two printers are identical.

New features of the 3211 include a powered forms stacker, an automatic platen, and a tapeless carriage. The powered stacker mechanism is self-adjusting and automatically rises in increments as the stack of paper mounts. This ensures that the stacker mechanism is always the same distance above the top of the stack of forms. The rate of rise during each increment is determined by the setting of the stacker rate knob, which can be adjusted by the operator to produce the best condition for the thickness of the forms being stacked. The stacker also can be raised or lowered manually.

When forms are inserted, the printer platen automatically positions itself close to the train cartridge in accordance with the thickness of the forms. Thus, correct clearance between the platen and the cartridge is achieved without operator intervention. Because of its automatic forms thickness sensing, the 3211 is sensitive to forms with a different degree of thickness at each edge. (For forms limitations, see Form-Design Considerations--System Printers.)

Forms control paper carriage tape loading and unloading by the operator is eliminated by implementation of a tapeless carriage feature for the 3211. Forms spacing and skipping are controlled by a program-loaded forms control buffer (FCB) contained in the 3811 control unit.

The FCB contains 180 storage positions, each of which corresponds to a print line, that is, a single space of the carriage. Thus, forms up to 22.5 inches in length can be accommodated at eight lines per inch spacing (or 24 inches at six lines per inch). Up to twelve channel codes (1-12), corresponding to the twelve channel positions of the paper carriage tape used on a 1403 Printer, can be stored in the appropriate buffer line positions to control carriage skipping. The FCB can be considered to contain a storage image of a carriage control tape.

A carriage control address register is used to address the FCB and maintain correct line position with respect to the form. This register is incremented as space and skip commands, which cause the form to advance, are issued. When a SKIP TO CHANNEL command is executed, the carriage control address register is incremented and the form moves until the channel specified is sensed in a line position in buffer storage. If the requested channel number is not found in the buffer, forms movement stops after address position 1 (line 1) has been sensed twice. This prevents runaway forms skipping.

A flag in a buffer storage line position is used to indicate the last line of the form for forms shorter than 180 lines. A flag bit is also used in the first buffer storage position to indicate six or eight lines per inch spacing. The FCB is loaded with the desired forms spacing characters via a LOAD FCB command issued by a program. An error indication is given if an end-of-page flag is not present or if an invalid carriage code is loaded.

Serviceability features, in addition to those provided for the 1403 Printer, are incorporated into the design of the 3211. The fact that a 3811 control unit controls only one 3211 Printer, instead of multiple devices, permits offline repair of the malfunctioning printer or control unit only, without the necessity of removing other operational units from the system.

The 3811 control unit presents six bytes of sense information to identify printer and control unit malfunctions instead of only one byte, as is provided for the 1403. Certain errors (such as a parity check in the print line buffer) that might be corrected by programmed retry of the print operation are identified in the sense bytes, and carriage motion is suppressed. This permits error recovery without operator intervention if the retry is successful. The additional

status data presented can be stored for later analysis and should speed the diagnosis of hardware malfunctions.

20:20 THE 3803/3420 MAGNETIC TAPE SUBSYSTEM

The new 3803/3420 Magnetic Tape Subsystem consists of 3803 Tape Control and a family of three 3420 Magnetic Tape Units, which read and write nine-track, 1600-BPI, phase-encoded, half-inch magnetic tape. The three tape units, Models 3, 5, and 7, have a data rate of 120 KB, 200 KB, and 320 KB, respectively, and up to eight tape units, in any mixture of models, can be attached to a 3803 control unit. This tape subsystem, which embodies a completely new control unit technology, offers price performance improvements, compatibility with existing seven- and nine-track tape volumes and programs, enhanced reliability, availability, and serviceability features, lower cost tape-switching capabilities, and standard automated tape-handling features presently available only on 2420 Magnetic Tape Units. (Table 20.20.2 at the end of this subsection compares 3420 and 2401 tape unit characteristics.)

The 3803/3420 subsystem can be attached to all System/370 models and to System/360 Models 30 to 195 (Model 67 in 65 mode only), except Model 44 for which there is no program support. The tape commands, status responses, and basic sense data of this tape subsystem are compatible with those of 2400-series tape units. Thus, any correctly written, non-time-dependent System/360 program for 2400-series tape units will operate without change on the Model 135 (subject to restrictions stated in Section 10:05) to handle operations on 3803/3420 subsystems with equivalent features installed. That is, existing nine-track 1600-BPI phase-encoded (PE), nine-track 800-BPI non-return-to-zero (NRZI), and seven-track 556/800-BPI NRZI-encoded tapes can be processed on 3420 tape units using existing programs without change to the tape volumes or programs.

The 3803/3420 tape subsystem offers users with intermediate systems the advantages of the latest significant advances in tape speed and design while maintaining media compatibility with existing tape volumes and providing enhanced RAS features. Specifically, the following are provided:

- Data rates of 120 KB, 200 KB, and 320 KB at 1600-BPI density
- Phase-encoded data recording that automatically detects and corrects single-bit read errors in-flight
- A tape transport design that minimizes tape wear and increases reliability, a single-drive capstan to control tape movement that provides faster data access times and rewinds, and more precise control of motor speed to help minimize damage to tape media
- Cartridge loading of tape, automatic tape threading, and a new automatic tape reel hub latch, all to reduce tape setup time
- Dual Density and Seven-Track (mutually exclusive) features to enable a 3420 tape unit to handle either nine-track 800-BPI NRZI and 1600-BPI PE tape or seven-track 556/800-BPI NRZI (BCD or binary) tape
- Flexible, lower cost tape switching implemented in a new compact physical design. A two-channel switch is available also.
- Features such as new technology to improve subsystem reliability and new diagnostic facilities to aid serviceability and thereby increase subsystem availability

Phase-encoded recording. The phase-encoded (PE) recording technique offers superior error detection and reliability as compared with the non-return-to-zero (NRZI) technique. In both cases, magnetic recording of one and zero bits is accomplished by means of flux reversals or changes in polarity. In NRZI recording, only one bits are recorded as magnetized spots, and a flux reversal occurs only for one bits. In PE recording both zero and one bits are recorded (the zero bit and one bit being opposite in polarity), and a flux reversal is required in every bit position. Thus, the PE dual flux recording technique differentiates between no recording and the presence of a zero bit, and the absence of any signal is detected as an error.

The positive recording of all zero and one bits in PE eliminates the need for horizontal parity bits (longitudinal redundancy check used in NRZI recording), and vertical parity bits are used to correct single-bit read errors in-flight. During reading, if a single track fails to respond with a suitable pulse in any bit position, reading of the rest of that track is immediately disabled for the remainder of the data block, and the remaining bits for that track are automatically generated by use of the vertical parity bits. In-flight single-track error correction eliminates the time normally lost in backspacing and rereading NRZI tape for correction of single-track dropouts or defects.

Phase encoding offers other advantages. If a string of zeros is recorded on tape, successful reading in NRZI requires close synchronization to "count" the correct number of zeros. With PE, this synchronization is provided by the flux reversal in every bit position; hence, PE recording (and reading) is self-clocking. In addition, each block written on a PE tape is preceded and followed by a coded burst of bits in all tracks to set up the individual track-clocking rates. The read circuitry is designed to recognize these bursts and thereby minimize the effect of noise in the gap.

The critical nature of vertical skew (alignment of bits within a byte) that is imposed by NRZI recording is minimized by this individual track-clocking scheme (one clock per track versus one clock for the entire tape subsystem), and by the use of one-byte (nine-bit) capacity skew buffers that can be in the process of collecting up to four data bytes at the same time, as the tape passes the read head. Because of the positive recording of all bits, once a skew buffer contains nine bits, one from each horizontal data track, it is an indication that a byte has been read. Thus, the 3420 can handle the situation in which the tape is not exactly aligned, and bits from up to four adjacent bytes can be read concurrently.

Like 2400-series tape units, the 3420 utilizes a two-gap read/write head that performs readback checking during write operations. The 3420 also has a separate erase head that erases the entire width of the tape during any write operation before writing occurs. Full-width erasure reduces the likelihood of leaving extraneous bits in interblock gaps or skip areas and minimizes the interchangeability problems that can occur when tape is written on one tape unit and read on another.

Advanced engineering design. The tape path in the 3420 tape unit is designed for "soft handling" of tape volumes to minimize tape wear and thus improve tape reliability. Other features, such as the single-drive capstan and optical tachometers, result in faster data access and rewind times than those of the 2401.

On a 3420 tape unit, the tape reel is mounted on the right side of the tape transport, instead of on the left as on a 2401 tape unit, so that an inverted tape path exists. As a result, when the tape is loaded in the columns, the recording side touches only the tape cleaner

and read/write head. Friction and tape wear are also reduced by the presence of air bearings in the tape transport that provide a thin film of air between the nonrecording surface and each metal bearing.

Use of a single-drive capstan transport for tape movement and optical tachometers for control of motor speed result in several advantages. First, faster access times than those of 2401 tape unit models are achieved. Access time is defined as the time interval from initiation of a write or forward read command (given when the tape is not at load point) until the first data byte is read or written, assuming the tape is brought up to speed from stopped status. Nominal access times for 3420 Models 3, 5, and 7 are 4.0 ms, 2.9 ms, and 2.0 ms, respectively.

Second, the single-drive capstan can be made to operate faster than normal read/write speed, and in-column rewind is thus implemented. Full reel rewind speeds average 410, 480, and 640 inches per second for Models 3, 5, and 7, respectively. In addition, less time is required to rewind less than a full reel on a 3420 as compared to a 2401 because of faster rewind times achieved by in-column rewinding.

Last, three optical tachometers that monitor motor speed are used to achieve precise control of the speed of both the capstan motor and the tape reel motors. The capstan tachometer measures the size of the interblock gaps (IBG's) created during tape writing. The result is a more consistent IBG size (.6 inches) than is created by 2400-series tape units, which enables more accurate calculation of tape passing time. IBG passing times are 8.0 ms, 4.8 ms, and 3.0 ms for 3420 Models 3, 5, and 7, respectively. These times would be used in calculations for command chained tape operations (reading or writing more than one tape block with a single START I/O instruction). More precise capstan motor speed also results in smoother starts and stops, thereby minimizing tape stretching and breaking.

The two tape reel tachometers measure tape speed as the tape enters and leaves the vacuum columns, and tape speed is adjusted when necessary. The 3420 tape unit is, therefore, less sensitive to voltage changes. More precise control of tape reel motor speed improves rewind speed and minimizes erratic tape stacking during rewinds so that there is less chance of damaging tape edges.

Automatic threading and cartridge loading. These advanced features are standard on all 3420 models and significantly reduce tape mounting and demounting time. Tape threading is automatic for tape reels not enclosed in a wraparound cartridge once the reel (10.5-inch, 8.5-inch, or minireel) is mounted on the tape unit with the tape end placed in the threading chute and the load-rewind button is depressed. The power window is closed, the tape is threaded on the takeup reel, and the tape is loaded in the columns and positioned at load point within ten seconds after the button is depressed for Models 3 and 5. On the Model 7, only seven seconds are required. In addition, unload and rewind/unload operations cause the tape to be completely rewound on the tape reel and the power window to be lowered so that the reel is ready for immediate demounting.

If the tape is enclosed in a wraparound cartridge (10.5-inch reels only), an operator need only mount the cartridge and does not have to place the tape end on the threader chute. Once the load-rewind button is depressed, ten seconds are required to open the cartridge and perform automatic threading. If automatic threading fails on the first try, the 3420 unit automatically rewinds the tape and retries threading. Unload operations rewind and close the cartridge automatically. In addition to fast tape reel mounting, the use of a wraparound cartridge offers other advantages. Handling of the tape reel itself is not required when the tape is used, because the

wraparound cartridge is also the shelf storage container. The only time the cartridge need be opened is when it is opened by the 3420 during use. This enhances the reliability of the tape media.

The 3420 tape unit also has a new automatic reel latch instead of the snap type hub latch implemented on newer 2400-series tape units. The operator places the tape reel on the hub and the automatic latch mechanically aligns the reel and then pneumatically locks it in position.

The advantage of these features can be shown by comparing setup times for tape units with and without the autothread feature. A tape study using experienced operators indicated the total time required to remove a tape reel, mount a new reel, thread the tape, and come to ready status was the following:

2401 tape unit - 40 seconds
Autothread tape unit without cartridge - 29 seconds
Autothread tape unit with cartridge - 13 seconds

Single Density, Dual Density, and Seven-Track features. These three features are provided for both 3803 control units and 3420 tape units. They are mutally exclusive features. Dual Density can be field installed on a 3803; however, the Seven-Track feature is not recommended for field installation. The Dual Density or the Seven-Track NRZI feature can be field installed on a 3420 tape unit only if it is replacing another NRZI feature. (For example, Dual Density can be field installed to replace the Seven-Track but not the Single Density feature.) The Dual Density and Seven-Track features facilitate efficient conversion of existing NRZI-recorded tapes to 1600-BPI phase-encoded format and permit tape volume interchange with other systems that use seven-track 556/800-BPI or nine-track 800-BPI tape. (See Section 60 for a discussion of conversion to 3420 tape units.)

A 3803 control unit with the Single Density feature (without a switching feature) can handle up to eight 3420 tape units (Models 3, 5, and 7) with the companion Single Density feature installed. Only 1600-BPI PE tape can be read and written. When the Dual Density feature is present on the 3803 control unit, both nine-track 1600-BPI PE and nine-track 800-BPI NRZI tape operations can be performed on 3420 units (Models 3, 5, and 7) with the companion Dual Density feature installed. (Tape units with the Single Density feature still handle only nine-track 1600-BPI PE tape.) When the Seven-Track feature is present on the 3803 control unit, seven-track 556/800-BPI NRZI operations (both BCD and binary format) can be performed on 3420 tape units (Models 3, 5, and 7) with the companion Seven-Track feature installed. The data convert and translate facilities are a standard part of the Seven-Track feature. Table 20.20.1 summarizes 3803 control unit capabilities without and with these features.

Tape mode setting is handled as follows. For write operations on nine-track tape units with the Dual Density feature, a mode set command must be issued to establish 1600-BPI PE or 800-BPI NRZI recording mode prior to the first write. Tapes written in PE mode have a format identification burst recorded at load point that differentiates them from NRZI mode tapes. During reading, sensing of this burst automatically puts the tape unit in PE mode. Failure to sense the burst establishes NRZI mode if both the tape unit and control unit have the Dual Density feature. If an attempt is made to read NRZI-mode tape on a unit without the Dual Density feature, an error indication results. Once PE or NRZI mode is established for read operations, it is retained until the tape returns to load point.

For seven-track read and write operations, NRZI mode, density, parity, and use of the data converter or translator are established by issuing a single MODE SET command.

Table 20.20.1. 3803 control unit configurations and capabilities with Single Density, Dual Density, and Seven-Track features

3803 with Single Density Feature	3803 with Dual Density Feature	3803 with Seven-Track Feature (includes data convert and translate)
1. Nine-track, 1600-BPI, PE tape on 3420 Models 3, 5, and 7 with Single Density feature	1. Nine-Track, 1600-BPI, PE tape on 3420 Models 3, 5, and 7 with Single Density feature	1. Nine-Track, 1600-BPI, PE tape on 3420 Models 3, 5, and 7 with Single Density feature
	2. Nine-track, 800-BPI NRZI tapes and nine-track, 1600-BPI PE tapes on 3420 Models 3, 5, and 7 with the Dual Density feature	2. Seven-track, 556/800-BPI, NRZI BCD and binary tapes on 3420 Models 3, 5, and 7 with the Seven-Track feature
Note: The Single Density, Dual Density, and Seven-Track features are mutually exclusive on the same control unit or the same tape unit.		

Tape-switching features. Tape subsystem configuration flexibility is provided by field-installable tape-switching options that permit up to four control units to be switched among up to 16 tape units. While this capability is provided for 2400-series tape units via the 2816 Switching Unit, tape switching for the 3803/3420 subsystem offers the advantages of compact design, reduced cost, and enhanced subsystem availability.

The switching features are built into the 3803 control unit itself so that space for stand-alone switching units is not required. The fact that tape-switching features are contained in the 3803 control units being switched (rather than in one unit) also enhances tape subsystem availability. When a switch failure occurs in one control unit, that unit can be switched offline, eliminating the necessity of removing the entire tape-switching subsystem from the operative system configuration.

Using combinations of the Communicator and the Two-Control Switch, the Three-Control Switch, or the Four-Control Switch optional features, two, three, or four control units can be configured to be switched among up to 8 or up to 16 tape units. The Communicator must be present in all control units that are to be switched. It allows the control unit in which it is installed to address tape units that are attached to an interconnected control unit. Figure 20.20.1 shows the switching feature requirements for permissible switching combinations. The switch combinations shown for switching control units among up to 16 tape units are the same that are required for switching control units among up to 8 tape units.

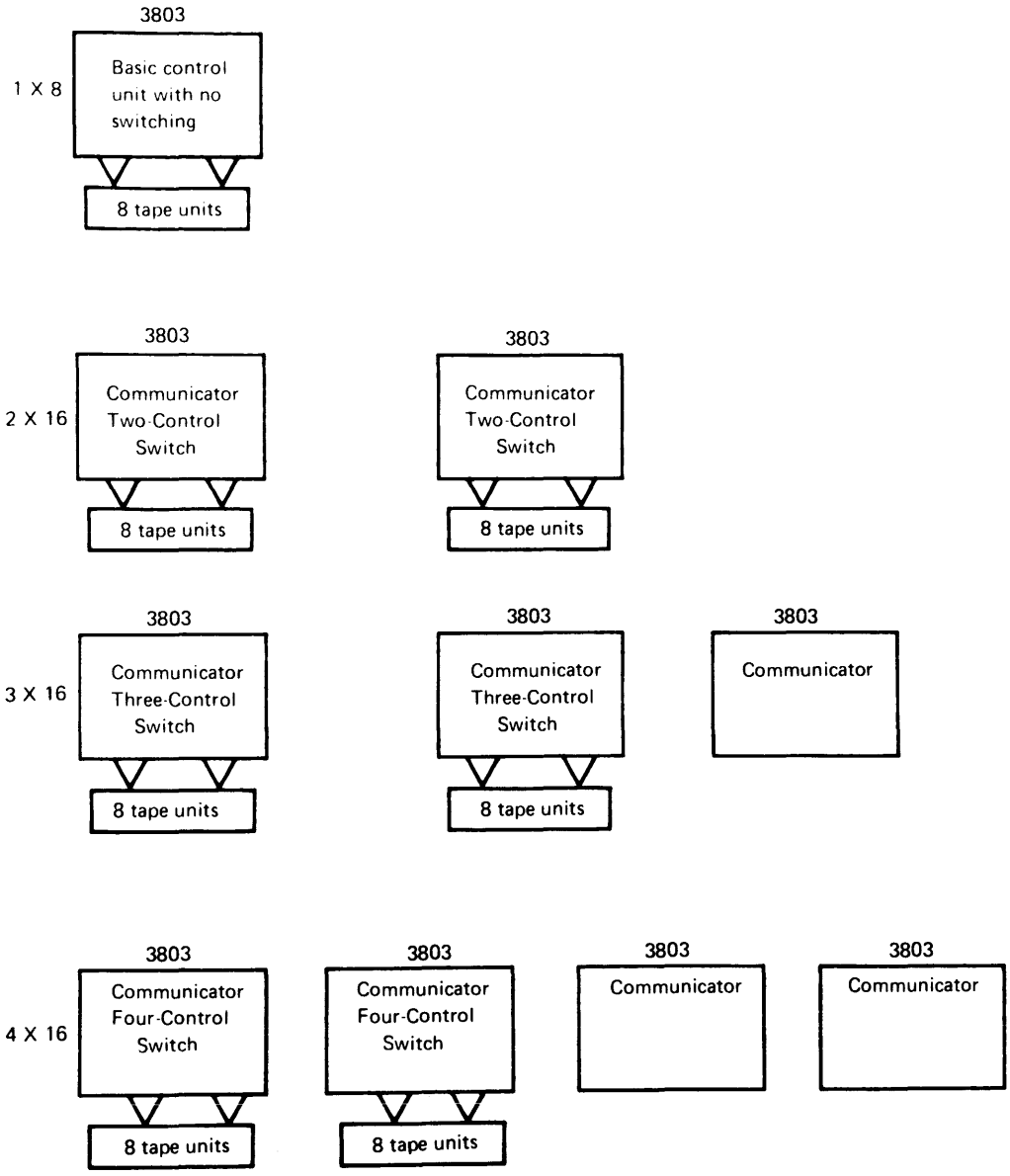


Figure 20.20.1. Tape-switching configurations for the 3803/3420 Magnetic Tape Subsystem

A two control unit switching configuration is required to replace the 2804 and 2404 read-while-write control units. The advantage of the tape switching approach is that for a small price increment better performance is possible. This is true because any two tape operations can be active concurrently in a switched configuration (including two reads or two writes) while the degree of simultaneity achieved using a read-while-write control unit is application dependent. That is, the application must lend itself to reading, then writing (or vice versa).

Two-Channel Switch. This optional feature provides switching functions for tape units similar to those provided by the two-channel switch for direct access devices. A 3803 control unit with the two-channel switch installed can be attached to two channels that are in

the same system or in two different systems. Tape units attached to the 3803 can then be accessed via either channel. This feature can be present on a 3803 that also has tape-switching features installed.

A 3803 with this switch can be set to allow access to all its tapes by either channel, one channel at a time. If channel A requests an operation when the control unit is busy performing an operation on channel B, channel A must wait until the control unit becomes available again. If both channels are on the same system, this arrangement essentially provides two channel paths to the tape units on the switched 3803.

A RESERVE CONTROL UNIT command is provided for use with this feature. It permits a channel, via programming, to maintain exclusive use of the control unit and its tapes until the RELEASE CONTROL UNIT command is issued. These two commands are of benefit when the control unit is shared between two systems.

Tape units on a 3803 with a two-channel switch can also be partitioned. Partitioning is the manual assignment of tape units (via switches) to one channel or the other so that access to each unit is limited to one of the two channels. This facility can be used for backup purposes to switch tapes from one system to another or from one channel to another in the same system.

The two-channel switch for the 3803/3420 subsystem offers configuration flexibility not generally available to 2400-series tape unit users. A two-channel switch currently is provided only for a 2803 Model 1 control unit and can be used only in Model 67 and in Model 65 multiprocessing configurations.

Reliability, availability, and serviceability features. The 3803/3420 hardware subsystem has several RAS features, in addition to the reliability and availability features already discussed for the tape media itself.

The 3803 control unit embodies a totally new design. The newest monolithic logic technology is used in the 3803 control unit and, therefore, it offers greater reliability and more compact physical design in comparison to the 2803 control unit. (The 3803 is approximately half the size of a 2803 control unit.) In addition, both logic circuitry and mechanical components in the control and tape units are functionally packaged to enable more rapid fault location and faster replacement.

As a diagnostic aid, additional sense bytes are generated by the microprogram-controlled 3803 control unit. The 3803 uses ROS for microprogram residence. Twenty sense bytes are provided, instead of the six generated by the 2803, certain of which can be used in tracing control unit microprogram malfunctions. Some of the other additional sense bytes identify the control unit and tape unit by serial number, optional features, and engineering change (EC) level.

Two other very significant new serviceability features are microdiagnostics resident in the 3803 control unit and radial attachment of 3420 tape units to the 3803.

Resident microdiagnostics in the 3803 enhance test operations for the 3803/3420 subsystem by relieving the CPU of the execution of most time-dependent tests. Diagnostics in the 3803 are executed via use of a diagnostic command issued by a program.

The 3803 also contains diagnostics that are operative during normal tape processing operations. These diagnostics perform operations such as the monitoring of measurement functions of the tape units. If an

irregularity is noted, the control unit generates sense bits to inform the executing program of the malfunction.

Tape subsystem availability is improved by radial attachment of 3420 tape units to the 3803 control unit. That is, each 3420 is cabled directly to the control unit so that any malfunctioning tape unit can be disconnected from the tape subsystem for servicing without disturbing the other tape units. When tape units are attached to the control unit in series (each tape unit cabled to the next tape unit), as are 2400-series units, the entire tape subsystem must be taken offline to uncable a tape unit.

These new features, combined with the use of fewer adjustable parts, are designed to provide optimum tape subsystem availability through better reliability and reduced maintenance time.

In conclusion, the 3803/3420 Magnetic Tape Subsystem offers Model 25 and 30 users of 2401 tape units the following advantages:

- Increased throughput for tape operations because of faster data rates, faster access times, and less rewind time for short files. In-flight correction of single-bit read errors eliminates a backspace and reread procedure and reduces the number of permanent read errors.
- Reduced tape setup time because of automatic tape threading and cartridge loading
- Reduced tape library size because of 1600-BPI density
- Less tape media wear as a result of the transport design and automatic threading and less tape damage caused by handling if wraparound cartridges are used for tape volume mounting and storage
- Reduced maintenance time because of the transport design (fewer adjustable parts), functional packaging of components, expanded sense bytes, and microdiagnostics resident in the control unit
- Increased tape subsystem availability because of reduced maintenance requirements
- Compatibility with existing 2400-series tape volumes and programs

These advantages, combined with lower subsystem cost and compact, flexible tape-switching capability, make the 3803/3420 Magnetic Tape Subsystem the natural growth path for tape users.

Table 20.20.2. 3420 and 2401 Magnetic Tape Unit characteristics

Characteristic	3420 Tape Units			2401 Tape Units					
	Model 3	Model 5	Model 7	Model 1	Model 2	Model 3	Model 4	Model 5	Model 6
Data rate (KB)	120	200	320	30	60	90	60	120	180
Density (bytes/inch)	1600	1600	1600	800	800	800	1600	1600	1600
Tape speed (inches/sec.)	75	125	200	37.5	75	112.5	37.5	75	112.5
Nominal interblock gap size in inches (nine-track)	.6	.6	.6	.6	.6	.6	.6	.6	.6
Nominal read access to data (ms)	4.0	2.9	2.0	16	8	5.3	16	8	5.3
In-column rewind	Yes	Yes	Yes	No	No	No	No	No	No
Nominal rewind and unload time (secs.)	76	66	51	132	90	66	132	90	66
Nominal rewind to ready status--full 2400-foot reel (secs.)	70	60	45	180	84	60	180	84	60
Automatic threading	Standard	Standard	Standard	Not available	Not available	Not available	Not available	Not available	Not available
Time to ready status after load button pressed (secs.)	10	10	7	-	-	-	-	-	-
Cartridge loading (10.5-inch reels only)	Standard	Standard	Standard	Not available	Not available	Not available	Not available	Not available	Not available
Automatic reel latch	Yes	Yes	Yes	No	No	No	No	No	No
Recording technique	PE	PE	PE	NRZI	NRZI	NRZI	PE	PE	PE

Table 20.20.2. 3420 and 2401 Magnetic Tape Unit characteristics (continued)

Characteristic	3420 Tape Units			2401 Tape Units					
	Model 3	Model 5	Model 7	Model 1	Model 2	Model 3	Model 4	Model 5	Model 6
Recording medium (1/2-inch magnetic tape)	IBM Series/ 500 Dynexcel, Heavy Duty, or equivalent 10.5",8.5", 6.5" reels. (Use of Mylar* is not recommended.)	Same as Model 3	Same as Model 3	Same as 3420 plus Mylar	Same as Model 1	Same as Model 1	Same as 3420	Same as 3420	Same as 3420
Inverted tape path, single-capstan drive optical tachometers	Yes	Yes	Yes	No	No	No	No	No	No
Error checking									
Single-track corrections during reading	Automatic	Automatic	Automatic	Programmed	Programmed	Programmed	Automatic	Automatic	Automatic
Vertical redundancy check	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Longitudinal redundancy check	No	No	No	Yes	Yes	Yes	No	No	No
Number of sense bytes	20	20	20	6	6	6	6	6	6
Microdiagnostics in control unit	Yes	Yes	Yes	No	No	No	No	No	No
Separate erase head	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Seven-Track feature	Optional	Optional	Optional	Optional	Optional	Optional	Not available	Not available	Not available
Densities (BPI)	800 556 -	800 556 -	800 556 -	800 556 200	800 556 200	800 556 200	- - -	- - -	- - -

*Trademark of E. I. Dupont de Nemours & Co. (Inc.)

Table 20.20.2. 3420 and 2401 Magnetic Tape Unit characteristics (continued)

Characteristic	3420 Tape Units			2401 Tape Units					
	Model 3	Model 5	Model 7	Model 1	Model 2	Model 3	Model 4	Model 5	Model 6
Data rate (KB)									
800 BPI	60	100	160	30	60	90	-	-	-
556 BPI	41.7	69.5	111.2	20.8	41.7	62.5	-	-	-
200 BPI	-	-	-	7.5	15	22.5	-	-	-
Recording technique	NRZI	NRZI	NRZI	NRZI	NRZI	NRZI	-	-	-
IBG size (inches)	.75	.75	.75	.75	.75	.75	-	-	-
Translator	Standard	Standard	Standard	Standard	Standard	Standard	-	-	-
Data Converter	Standard	Standard	Standard	Optional	Optional	Optional	-	-	-
Dual Density feature (800/1600 BPI)	Optional	Optional	Optional	Not available	Not available	Not available	Optional	Optional	Optional
Data rate (KB) at 800 BPI	60	100	160	-	-	-	30	60	90
Recording technique at 800 BPI	NRZI	NRZI	NRZI	-	-	-	NRZI	NRZI	NRZI
IBG size at 800 BPI (inches)	.6	.6	.6	-	-	-	.6	.6	.6
Control unit	3803 with optional Seven-Track or Dual Density feature (not both). Read while write (RWW) capability is not provided	Same as Model 3	Same as Model 3 but no Seven-Track feature	2803, 2804 (RWW) Model 1 with optional Seven-Track Compatibility feature 2803, 2804 Model 2 with optional Seven-Track, Nine-Track, or Seven- and Nine-Track Compatibility feature	Same as Model 1	Same as Model 1	2803, 2804 (RWW) Model 2 with optional Seven-Track, Nine-Track, or Seven- and Nine-Track Compatibility feature	Same as Model 4	Same as Model 4

Table 20.20.2. 3420 and 2401 Magnetic Tape Unit characteristics (continued)

Characteristic	3420 Tape Units			2401 Tape Units					
	Model 3	Model 5	Model 7	Model 1	Model 2	Model 3	Model 4	Model 5	Model 6
Tape switching	2 x 16 3 x 16 4 x 16 (Switching features in 3803 control units)	Same as Model 3	Same as Model 3	2 x 16 3 x 16 4 x 16 (Requires one or two 2816 units)	Same as Model 1	Same as Model 1	Same as Model 1	Same as Model 1	Same as Model 1
Two-Channel Switch	Optional	Optional	Optional	Optional on 2803 Model 1 for Model 67 and MP65 systems only.	Same as Model 1	Same as Model 1	Not Available	Not Available	Not Available

20:25 THE 3505 CARD READER AND THE 3525 CARD PUNCH

These new 80-column card units provide a wide range of functional capabilities and high reliability. They are designed to minimize and simplify operator intervention. New functions such as optical mark reading, read column eliminate, and card printing, new error recovery features such as automatic punch retry and feed retry, and new diagnostic procedures are provided.

The 3505 Card Reader, available in two models, provides medium- (800 cpm) and high-speed (1200 cpm) card reading, standard column binary read capability, and optional optical mark reading. The 3525 Card Punch, available in three models, offers a range of punch speeds (100, 200, or 300 cpm) as well as optional card reading and card printing. Complete configuration flexibility is provided. Any speed model of the punch, with any combination of its options, can be combined with either speed model of the reader, with any combination of its options, to provide the speeds and functions desired by an installation.

The 3505 Card Reader, shown in Figure 20.25.1, attaches to System/370 Models 135 and up. The 3505 contains a control unit, which is fully buffered to prevent data overrun. The 3505 can be attached to a byte multiplexer, selector, or block multiplexer channel and can be assigned any priority. When the 3505 is attached to a byte multiplexer channel, data is transferred between the channel and the 3505 in one-byte (EBCDIC mode) or two-byte (binary mode) bursts. When the 3505 is attached to a selector or a block multiplexer channel, the entire data record is transferred in burst mode.

The 3525 Card Punch, shown in Figure 20.25.2, attaches to a channel via the 3505 Card Reader. The maximum distance between the two units is 20 feet. Only one 3525 punch can be attached via each 3505 reader, which must have a 3525 adapter installed. The 3525 adapter provides all the additional control unit hardware required for independent, fully buffered operation of the 3525. While only one control unit position on a channel is required (as for a 2540 Card Read Punch), two subchannels (on a byte or block multiplexer channel) and separate unit addresses are used. The 3505 and 3525 appear to the channel to be two logically and physically independent devices. Although the reader and the punch use the same control unit in the 3505, most failures can be diagnosed and repaired on one unit while the other continues to operate.

Table 20.25.1 lists the features of these new units. Table 20.25.3 at the end of this subsection compares the features of the 3505 reader and the 3525 punch with those of the 2540 reader/punch.

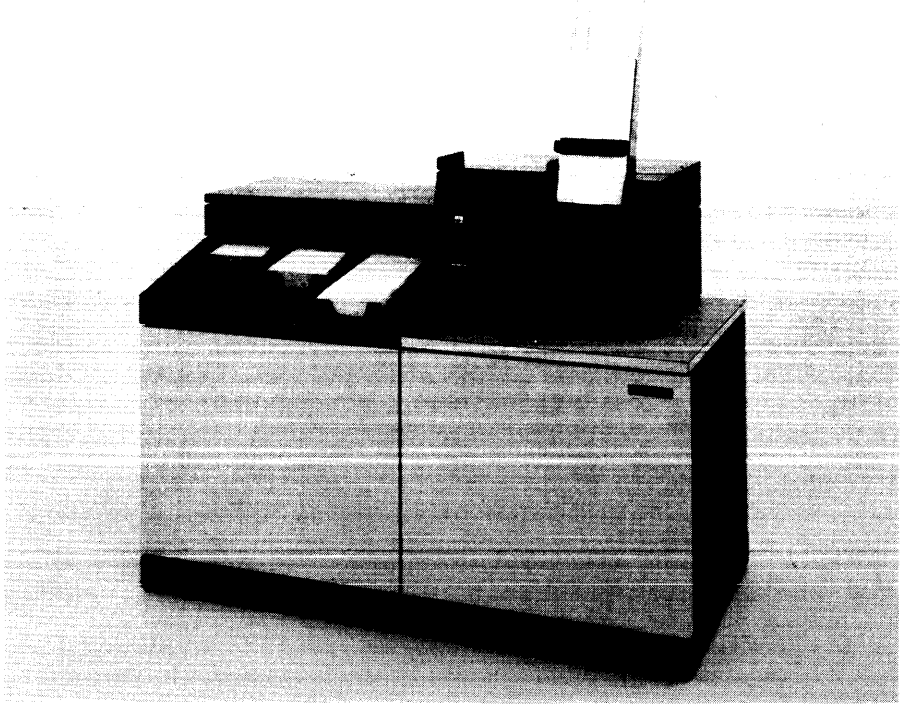


Figure 20.25.1. The 3505 Card Reader

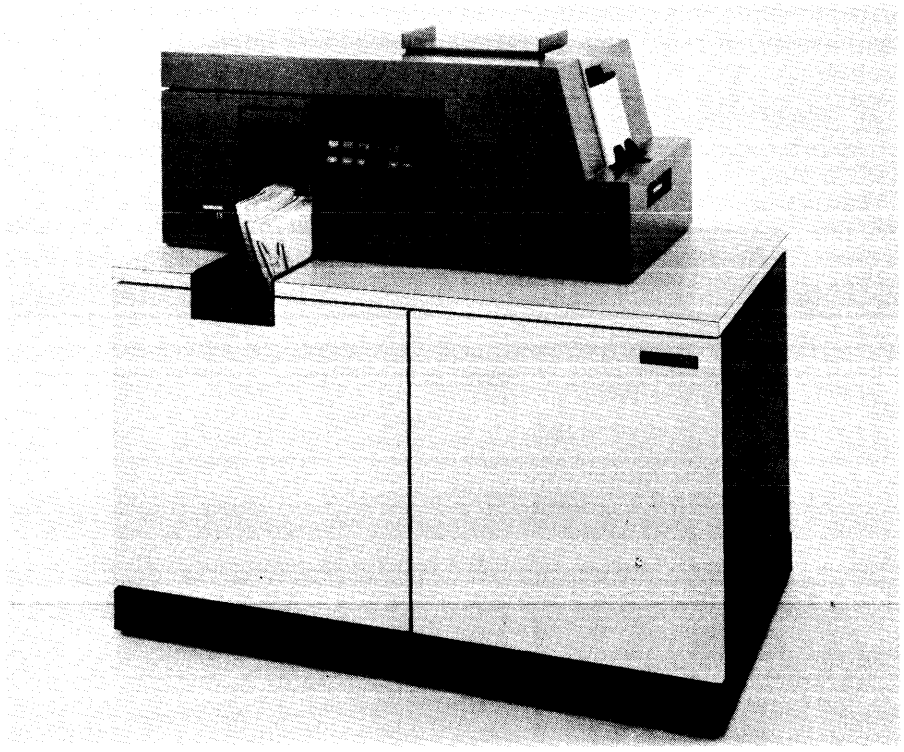


Figure 20.25.2. The 3525 Card Punch

Table 20.25.1. 3505 Card Reader and 3525 Card Punch features

Characteristic	3505 Card Reader	3525 Card Punch	3525 Card Punch with Card Read Option
Models and maximum rated speeds	B1 - 800 cpm read B2 - 1200 cpm read (Includes fully buffered control unit and attaches directly to a channel)	P1 - 100 cpm punch P2 - 200 cpm punch P3 - 300 cpm punch (Attaches to channel via 3505)	P1 - 100 cpm punch and read P2 - 200 cpm punch and read P3 - 300 cpm punch and read (Attaches to channel via 3505)
Standard features	• 3000-card-capacity file feed • One logical stacker consisting of two physical stackers with a 1750-card capacity each (alternate stacking feature) • Feed retry • Card Image	• 1200-card-capacity hopper • Two 1200-card-capacity stackers • Punch column binary • Automatic punch retry and dedicated error stacker	• 1200-card-capacity hopper • Two 1200-card-capacity stackers • Punch and read column binary • Automatic punch retry and dedicated error stacker (in non read/punch mode) • Read column eliminate
Optional features (field installable unless otherwise noted)	• Optical Mark Reading • Read column eliminate • Selective Stacker • 3525 Punch Adapter • 3525 Read Punch Adapter • 3525 Two-Line Print Control • 3525 Multi-Line Print Control • Special Feature Adapter (prerequisite for all other features)	• Two-Line Card Print (not recommended for field installation) • Multi-Line Card Print - up to 25 lines (not recommended for field installation)	• Two-Line Card Print (not recommended for field installation) • Multi-Line Card Print - up to 25 lines (not recommended for field installation)

THE 3505 CARD READER

Models B1 and B2 of the 3505 differ only in card reading speed, and model changes can be made at an installation. All optional features for the 3505 are also field installable. The Special Feature Adapter is a prerequisite for attachment to the 3505 control unit of any optional feature.

The I/O commands used for card reading on the 2501 and 2540 are a compatible subset of those used on the 3505. Additional commands are required for the 3505 to handle new features, such as read column eliminate and optical mark reading.

The 3505 offers several new features designed to enhance card reading. First, friction feeding with vacuum-assist is used to feed cards instead of the picker knives used on the 2540 because the latter becomes less effective as card reading speeds increase. When a feed operation is performed in the 3505, it causes the bottom card in the read hopper to drop down on a continuously rotating feed roll. The combination of feed roll motion and vacuum-assist causes the card to be fed to the first station. The use of friction feeding makes the 3505 less sensitive to feeding cards with frayed edges than if picker knives were used.

Card feeding, as implemented in the 3505, also provides another advantage. Feed control is completely asynchronous; there is no waiting for clutch-points as there is for the 2540. A feed is initiated as soon as the feed command is received. Therefore, the 3505 can achieve average card reading speeds that are closer to the maximum rated speed of the model when the time between feed commands exceeds the 75-ms or 50-ms cycle required to maintain 800- or 1200-cpm reading. For example, reading speed on a 2540, which uses a multi-toothed clutch in card feeding, drops from 1000 cpm to 750 cpm if a feed cycle occurs every 62 ms instead of every 60 ms. On a 3505 Model B2, reading speed averages approximately 1154 cpm, instead of 1200 cpm, if a feed cycle occurs every 52 ms instead of every 50 ms.

Second, an automatic feed retry function is implemented. If a card fails to feed on the first try, three feed retries are made by the 3505 before a misfeed indication is given. This feature reduces operator intervention. Should a card jam occur during card movement, easy access to the entire card path within the 3505 helps quick removal of affected cards.

Third, a photoelectric read head, designed to provide a high degree of read reliability, is used to read cards serially by column. Optical reading is inherently more reliable than reading via brushes because a photoelectric read head is less prone to damage and/or wear than are metal read brushes. The method used for read checking gives the 3505 an improved capability of reading card columns that are offpunched or misregistered. Sixteen time counts are taken for the reading of each card column. A column is read at four of the middle 16 time counts (5, 6, 9, and 10). The four readings are then inspected for a valid combination of data or no data impulses. Data validity checking, performed when EBCDIC data is being read, is the same as that for the 2540. (More than one punch in rows 1 through 7 constitutes an invalid EBCDIC character.)

Last, read column eliminate (RCE) is an optional programmable feature that permits the 3505 to ignore card columns that contain perforations or other internal scores that would normally cause a read check. The new WRITE RCE FORMAT command must be issued before card feeding and reading occurs. This command is provided to set up the format under which card columns will be read and to establish read column eliminate format mode of reading. It transfers 80 bytes of data to the control

unit, indicating which columns are not to be read. Blanks are transferred to processor storage for these columns when cards are read in format mode. FEED, READ only, and READ AND FEED commands with the new format mode bit on in the operation code must then be issued to cause read column eliminate to be effective. Reading in format mode continues until the first feed type command without the format mode bit on is issued or until unit exception status (end of file) on the 3505 is accepted by the system.

Card stacking has been designed to require operator intervention less frequently. Two 1750-card-capacity stackers that functionally provide one logical stacker with a 3500-card capacity are standard. This is achieved through implementation of an alternate stacking facility. Initially, cards are placed in the first (or right-half) stacker and a light on the operator panel is turned on to indicate that the right half is the active stacker. When the right-half stacker becomes full, the right active light is turned off, cards automatically begin stacking in the second (or left-half) stacker if it is ready, and the left active light is turned on. When the left-half stacker becomes full, stacking is automatically switched back to the right-half stacker if the operator has removed all the cards from the right-half stacker and set the stacker ready switch on the operator panel to indicate an empty right-half stacker condition. Otherwise, a stacker-full indication is given. Thus, more than two and one-half times the number of cards can be stacked in a 3505 as in a 2500-series card reader before operator intervention is required. If care is exercised, cards can also be removed from a stacker while cards are being placed in it, as can be done with 2500-series readers.

If the optional Selective Stacker feature is installed, cards can be directed under program control to a second logical stacker (third physical stacker) with a capacity of 1750 cards. Alternate stacking into logical stacker 1 (right and left half) occurs as already described. An unlimited amount of time is available in which to issue the stacker select command. During this time the card is positioned at a wait station. A 3505 without the Selective Stacker feature installed ignores stacker select commands and places all cards in logical stacker 1. If a stacker select command indicating pocket 3 is received by a 3505 with the Selective Stacker feature, the card is placed in stacker 2. An error indication is not given in either situation.

The standard Card Image feature for the 3505 provides the same function as the Card Image or the Column Binary feature for other readers. Read column eliminate can also be operative during column binary mode reading.

The optional Optical Mark Read (OMR) feature provides the 3505 with the ability to read up to a maximum of 40 columns of vertical marks on an 80-column card. Both vertical mark fields and punched-hole fields can be read in one pass of a card. Vertical marks can be preprinted on cards in a nonreflective ink. Alternately, marks can be made by hand with a No. 2 pencil or equivalent, thus eliminating the necessity of using special graphic pencils, as is required for mark sensing operations on the 514 Reproducing Punch and the 519 Document Originating Machine.

As shown in Figure 20.25.3, the center of a vertical mark column is coincident with a punch column. Vertical mark fields can begin at any column and can be intermixed with punched-hole fields in any sequence, subject to the following: Any vertical mark column must always be preceded by at least one blank column (except for column 1) and followed by at least one blank column (except for column 80). OMR card design, nonreflective ink requirements, marking restraints,

etc., are discussed in detail in IBM Component Description: 3505 Card Reader and 3525 Card Punch.

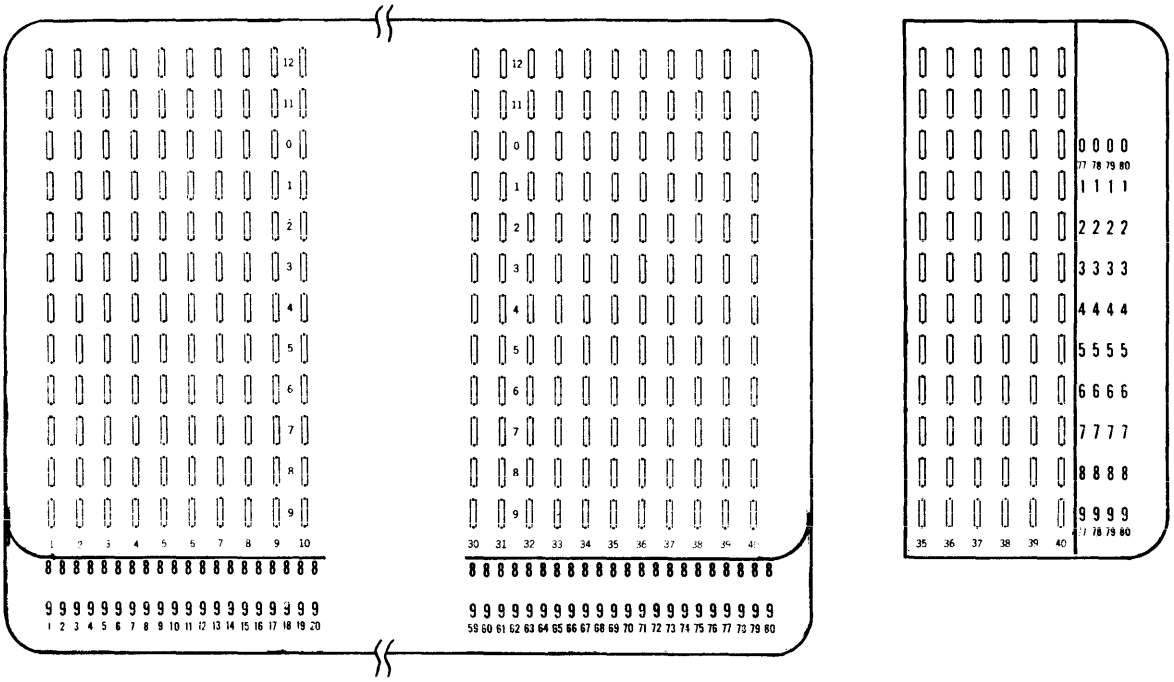


Figure 20.25.3. Vertical mark card format

OMR mode and format are established in the same manner as they are for read column eliminate. The new WRITE OMR FORMAT command must be issued to establish OMR format reading mode and to indicate which columns are to be read in OMR mode. FEED, READ only, and READ AND FEED commands must have the format mode bit on in the operation code to cause optical mark reading to be effective. OMR and read column eliminate cannot be used simultaneously (because they use the same format hardware and command mode bit).

Vertical mark fields can be read in EBCDIC or column binary mode and validity checking is performed on EBCDIC data as usual. A vertical mark field transferred to storage does not contain the interspersed blank columns contained in the card. The control unit compresses vertical mark fields into contiguous data fields by sending only vertical mark columns to storage. Thus, a three-character vertical mark field in card columns 1 to 6 appears in positions 1 to 3 of the input area in storage.

When a vertical mark column is read, the control unit distinguishes between a marginal mark and a valid mark. A marginal mark indication can result from a poor erasure, a short mark, or a poor marking device. If a vertical mark column cannot be read properly, a hexadecimal '3F' in EBCDIC mode, or '3F3F', in column binary mode, is placed in the input area in that column position. (Hexadecimal '3F' corresponds to a 7, 8, and 9 punch in a card column. This character should not be used for any other purpose on cards to be read in OMR mode.) Column 80 (or 160) will also contain a hexadecimal '3F' when a marginal mark is sensed in any column. An error indication is not given, so that the user program can scan vertical mark fields for '3F' characters (if column 80 or 160 contains a '3F') and take the recovery action desired. If the Selective Stacker feature is installed, vertical mark cards with read errors can be selected into stacker 2, unless card sequence must be maintained.

A vertical mark field must not contain any punches or nonreflective marks in addition to the actual vertical mark data. Thus, if a card is designed to have the characters of a vertical mark field handprinted as well as vertically marked (for verification purposes, for example), the handprinted characters must not be contained in any vertical mark field. Further, any nonreflective marks made on the card should be separated from vertical mark fields by a minimum of two columns and no writing should appear on the leading edge of the card before column 1.

OMR offers low-cost data entry and can be used in a variety of application areas. As a means of data recording, it has the following advantages:

- The only equipment required is an ordinary pencil
- Very minimal user training is required and errors can be corrected easily by proper erasure
- Data can be recorded at any location, quietly and without the necessity of an electrical or communications hookup
- An unlimited number of people can record data simultaneously

A 3505 with OMR does not provide all of the facilities of the 514 and 519, specifically, punching into marked cards and end printing, the latter as on the 519. However, the 3505 with OMR offers advantages over 514/519 mark sensing:

- Significantly greater reading speeds (up to 800 or 1200 cpm versus 100 cpm)
- Use of ordinary instead of special graphic pencils
- Up to 40 columns of data per card versus a maximum of 27
- Elimination of an offline processing operation

The availability of OMR on the 3505 provides the opportunity to (1) integrate existing 514/519 mark sensing processing into the normal stacked job operations and (2) install new applications that will benefit from the use of this method of data entry. Many of the applications for which OMR offers advantages are listed below.

Education
Test scoring
Surveys
Scheduling
Problem solving
Student registration
Attendence reporting
Library recording
Supplies requisitioning
Grade reporting

State and Local
License renewals
Assessors reports
Traffic surveys
Statistics
Police activities
Accident reports
Public utilities
Motor vehicle inspection

Communications
Toll billing

Manufacturing
Physical inventory
Inventory receipts
Inspection
Plant maintenance reporting
Accounts receivable

Distribution
Order Entry
Mail order
Physical inventory

Utilities
Meter reading

Medical
Admission data
Medical history
Menu selection
Test results
Nurses notes

Federal
Census data
Military logistics

Airlines
Surveys
Beverage control
Inventory control

Insurance
Premium notices

Finance
Loan payment coupons

Media
Subscription fulfillment
Direct mail advertising
Book/record club promotion
Market surveys
Membership accounting

Either the 3525 Punch Adapter or the 3525 Read Punch Adapter and the Special Feature Adapter must be installed on the 3505 in order to attach a model of the 3525 Card Punch to a system channel. These features can be field installed. Models P1, P2, and P3 of the 3525 differ only in speed, and model changes can be made in an installation. Any 3525 model can be combined with either 3505 model.

The I/O commands used for the punch side of the 2540 without punch feed read capability are a compatible subset of those provided for the 3525. (The special punch feed read command defined for the 2540 is not used on the 3525 and it will cause a command reject error on the 3525.) Additional commands for the 3525 support new features such as card printing and read column eliminate. Since a 3525 with the Card Read feature installed can execute the same commands as the 3505 reader (all features except optical mark reading), the 3525 can be used as a second or a backup card reader, if necessary.

In the 3525, cards move from the hopper to a parallel read station (dummy if the Card Read feature is not installed) and on to a parallel punch station at which punch checking occurs. Then cards move to a parallel print station (dummy if a card print feature is not installed), after which they are stacked into one of two standard 1200-card-capacity stackers under program control. Thus, reading, punching, and printing can occur for each card during one pass, a facility not available on other card units for System/370 models or on card units for System/360 Models 30 and up.

Card feeding on the 3525 is handled by picker knives as on a 2540 because this technique is required for parallel card reading operations when an additional aligning station is not used. Parallel rather than serial feeding is used in the 3525 to obtain the punch speeds available. As on a 3505, the entire card path in the 3525 can be exposed quickly if a card jam occurs.

A new method of punch checking, which senses punch displacements, is implemented in the 3525. Punch checking is achieved by monitoring the actual motion of each punch. The control unit compares the output from the sensors attached to each punch to the data sent to be punched and determines whether holes were actually punched and, if so, punched correctly. Each card is also checked for correct position and skew to ensure punch registration accuracy.

A significant new recovery feature of the 3525 is automatic punch retry when combined read/punch operations are not being performed. (Punch retry is effective during punch/print operations.) When a punch error is detected, the control unit directs the error card to a 200-card-capacity dedicated error stacker (stacker 3) located under the covers of the 3525 and causes the output data in the buffer to be repunched into the next card. If the retry is successful, the correctly punched card is sent to the error stacker and the stacker 3 light is turned on to indicate to the operator that cards are in the error stacker. (A correctly punched card is placed in the error stacker so that the customer engineer can compare it with the incorrectly punched card, which can be useful in identifying the punch malfunction.) Another card is then punched with the same data and stacked, after which normal operations continue. These punch retry operations are controlled entirely by the control unit. Therefore, punch retry is transparent to the channel and the program and does not require operator intervention. If a single punch retry is unsuccessful, punching stops and an error condition is indicated. When the operator runs out the cards, the second error card is placed in stacker 1.

Because of the way in which punch retry operates, punching into prepunched or serially numbered preprinted cards should not be attempted unless the Card Read feature is present. A combined read/punch operation must also be defined so that a READ followed by a WRITE AND FEED command sequence is issued by data management. This sequence causes punch retry to be bypassed if a punch error occurs. Otherwise, when a punch retry occurred, data meant to be punched only in card N would also be punched into cards N+1 and N+2, and card N+1 would be placed in the error stacker.

The optional Card Read feature (field installable) for the 3525 provides column binary operations as a standard feature and offers a function not available with the Punch Feed Read feature for the 2540. Read column eliminate, as already described for the 3505, is standard on a 3525 with the read feature. Therefore, perforated or internally scored cards can be read from a 3525 as well as from a 3505. In addition, cards are read optically on the 3525, and thus offpunching and misregistration of cards are not as significant a factor in successful card reading as they are when read brushes are used.

Either the Two-Line Card Print or the Multi-Line Card Print optional feature can be installed on the 3525 in addition to the options already discussed. Either the 3525 Two-Line Print Control or the Multi-Line Print Control feature and the Special Feature Adapter are required on the 3505 as well. The card print features are mutually exclusive and not recommended for field installation; however, a change from either feature to the other can be made at the installation.

Engraved type slugs, similar to those used on a 1403 Printer, are used in the print cartridge assembly in contrast to the wire matrix technique used in the 2560 Multi-Function Card Machine. The cartridge assembly provided for the 3525 contains 64 characters and an installation can order one of two character sets, EBCDIC (63 characters including blank) or ASCII (64 characters including blank). If the ASCII cartridge is used, the program is responsible for providing ASCII encoded data in storage (that is, the control unit will not convert EBCDIC mode data from storage to ASCII mode for printing). Black and purple ink ribbons are available.

The Two-Line Print feature provides the 3525 with the ability to print two lines of data on a card, up to 64 characters in length, during a single pass. Card throughput for two-line printing operations equal to the rated punch speed of the 3525 model can be obtained--100, 200, or 300 cpm. The first line of print is located along the top edge of the card and the second is between punch rows 12 and 11. This corresponds to lines 1 and 3 when 25 lines are used, as shown in Figure 20.25.4.

The Multi-Line Card Print feature permits the 3525 to print up to 25 lines on a card, 64 characters in length, during a single pass. Print lines are shown in Figure 20.25.4. They are .125 inches apart, and identical in location to the 25 line locations defined for card printing using the 2560 MFCM. The lines printed on any given card are determined by programming. A print line command supplies the number of the line on which printing is to occur as well as the data to be printed and normally causes the card to be positioned in the print mechanism at the line indicated. Thus different lines can be printed on different cards. (Note that lines must be printed in ascending line number sequence and thus a given line cannot be printed on twice in one pass.) For both card print features, the data printed on each line must be supplied by the program and can include data read from a prepunched card (if the read feature is present). There is no limit to the time taken between print commands.

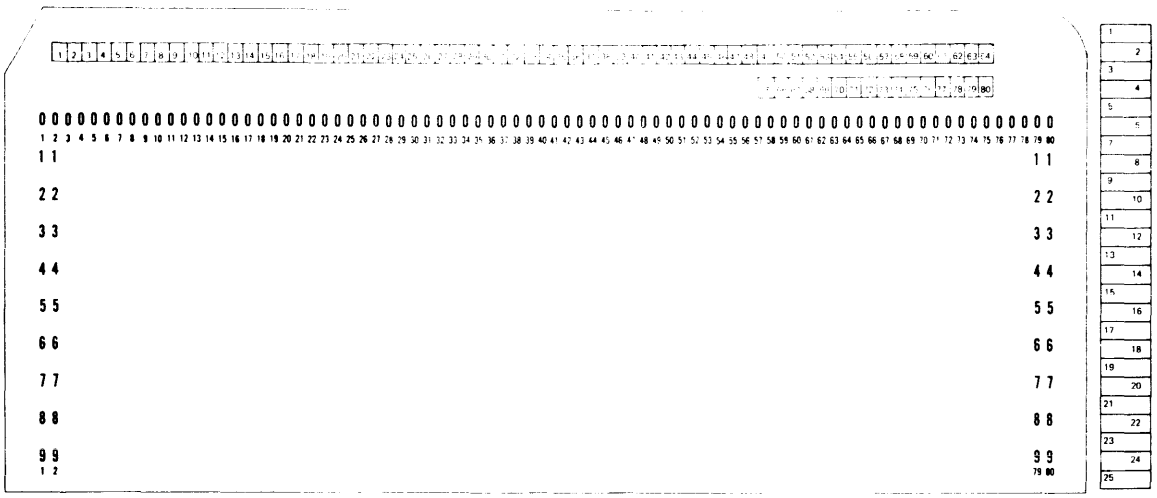


Figure 20.25.4. Card layout for 25 print lines

The card speed when printing on a given model of the 3525 with the Multi-Line Card Print feature is a function of the number of lines printed and their location. Some sample throughputs are shown below. Sixty-four characters are printed on each line.

Number of Lines Printed	Line Position	Speed in Cards per Minute		
		P1	P2	P3
1	1	100	200	300
2	1 and 3	100	200	240
3	11, 12, and 13	67	133	150
4	11, 12, 13, and 14	67	114	120
6	11 through 16	57	89	92
10	11 through 20	44	62	63
25	All	24	29	29

A technique called overlapping is used during print operations in order to increase card throughput. During printing operations on a card for all lines except the last two, the card is successively moved to the line position at which printing is to occur while other cards in the transport remain motionless. Printing of the last two, or only two, lines occurs during the next card feed cycle during which all cards in the transport move. In order to implement overlapping, the control unit stores the data for the last two lines to be printed (in buffers in the control unit) when the last two print commands are issued but does not move the card or print the lines. Thus, the time that would normally be taken to position the card and print the last two lines is saved.

Many of the application areas in which card printing is used are the following:

Manufacturing
Shop packet cards
Labor reporting
Material move control
Inventory issues and receipts
Physical inventory
Inspection
Plant maintenance reporting
Accounts payable
Accounts receivable

Insurance
Premium notices

Finance
Loan coupons
Christmas club

Utilities
Bills

Federal
Bonds
Notices

State and Local
Warrants
License applications
Vehicle titles
Record abstract requests

Media
Subscription fulfillment
Direct mail advertising
Book/record club promotion
Market surveys
Membership accounting

Airlines
Credit card billing
Cargo revenue accounting
Beverage control
Maintenance - shop control

Education
Test scoring
Attendance
Supplies requisitioning
Grade reporting
Class card preparation

Process
Petroleum billing

General
Proxy notices
Dividends
Program cards

Distribution
Accounts receivable

Table 20.25.2 compares the card print features offered by the 3525 Punch, the 557 Interpreter, the 1404 Printer, and the 2560 MFCM. Minimally, the card print capability of the 3525 can be used for card interpreting functions as a replacement for the 557. The 3525 offers increased printing throughput over the 557 because online printing on the 3525 eliminates a separate interpreting operation, multiple lines can be printed in one pass of the card, and one or two lines can be printed faster than on a 557. In addition, data not punched in the card can be printed on the 3525. The 3525 with card printing offers Model 25 users who are emulating Model 20 operations print capability in addition to that provided by the 2560 MFCM, which cannot be attached to the Model 135 (more flexibility in the lines printed during a pass, more than six lines printed per card pass, etc.) The 3525 with card printing capability can also be used as an alternative to 1404 cut card printing operations, which are not supported by OS or DOS data management.

Table 20.25.2. 3525, 557, 1404, and 2560 card-printing capabilities

Characteristic	3525 Punch with Card Print	557 Interpreter	1404 Printer	2560 MFCM
Systems to which attaches	All System/370 models	Stand-alone	System/360 Models 25 to 50	System/360 Models 20, 25
Operations possible on one pass	Read, punch, and print	Print data read	Print only	Read, punch, and print (also collate)
Programming support	OS and DOS	-	Cut-card operations not supported by OS and DOS	Not supported by OS and DOS
Maximum number print lines per card	2 or 25 (depending on print feature)	25	25	25

Table 20.25.2. 3525, 557, 1404, and 2560 card-printing capabilities (continued)

Characteristic	3525 Punch with Card Print	557 Interpreter	1404 Printer	2560 MFCM
Maximum number lines printed per pass of a card	2 or 25 (any number 1 to 25)	1	25 (any number 1 to 25)	6 (any number 1 to 6)
Number of characters per print line	64	60	69	64
Line selection	Programmed	Set by operator before processing begins	Programmed	Set by operator before processing begins
Character set size	63(EBCDIC) 64 (ASCII)	39 (EBCDIC)	Same as for 1403 Printer	63 (EBCDIC)
Speed (cards per minute)	Prints in parallel one line at a time. Speed when printing depends on number of lines printed and their location. Two 64-character lines can be printed at 100, 200, or 300 cpm with Two-Line Print feature. Up to 64 characters on lines 11 to 16 can be printed at 57, 89, or 92 cpm with Multi-Line feature.	100 for 1 line	Up to 800 with dual carriage	Prints serially one to six lines at a time. Speed when printing depends on the location of the last character printed and whether or not punching occurs on the same pass. Any six lines of 64 characters can be printed at 100 cpm.

ERROR RECOVERY PROCEDURES FOR THE 3505 AND 3525

Error recovery procedures have been redesigned to simplify, reduce, or eliminate operator intervention. The amount of programmed error recovery required is also reduced because of the many new functions implemented in the control unit in the 3505. These include the automatic feed retry and punch retry features already discussed.

The significant new feature of error recovery for these new units is that if the control unit itself cannot correct a failure, it

identifies not only the error but the specific recovery action to be taken by the error recovery procedure (ERP), the operator, or both.

When a unit error (unit check in the CSW) occurs, the control unit presents four sense bytes to the ERP (instead of one as on the 2540). The first two bytes specifically identify the error and indicate which of three possible recovery actions the ERP is to take. The second two sense bytes are provided for diagnostic purposes and are discussed under "Maintenance". The three programmed recovery actions defined are not device dependent, so that the same ERP is used for both the 3505 and the 3525.

The recovery actions to be taken by the ERP (as indicated in sense byte 1) are the following:

1. Post permanent error. The ERP posts the error as permanent without attempting any recovery. (An invalid command is an example of such an error.)

2. Retry once without operator intervention (called automatic retry). The ERP reissues the failing command once and normal processing continues if the operation is successful. If one retry is unsuccessful, permanent error is posted without further recovery attempts. (Since unusual command sequences, as defined for the 2540, are <u>not</u> defined for the 3505 and 3525, when an equipment check occurs, the failing command, say a read, can be reissued to attempt to correct the error. Equipment check is considered to be a permanent error on the 2540.)

3. Retry after operator intervention. When the unit is readied by the operator after the indicated recovery action has been performed, the ERP reissues the failing command. In this case, the operator could have attempted to correct the error or could have indicated that correction was not possible. If the error --a card jam, for example--was corrected, processing should continue normally when the failing command is reissued. However, if the operator determines that correction is not possible, such as when a card contains an invalid character, he can press the new permanent error key on the operator panel. When the ERP reissues the failing command after this key has been pressed, an error condition results as soon as the reissued command is received, and the appropriate sense byte is presented by the control unit to indicate that the operator pressed this key. Permanent error is then posted by the ERP without additional recovery action.

Implementation of the new permanent error key permits orderly abnormal termination of a job step to take place when certain uncorrectable failures occur. This key can also be used for operator communication with a user-written error routine that is entered after a permanent error occurs.

In order to pinpoint the specific recovery action to be taken when operator intervention is required, both the 3505 and the 3525 have a new backlighted panel as part of the operator panel. Buttons and lights on the respective operator panels are illustrated in Figure 20.25.5. The backlighted panel contains new lights, in addition to some of the same lights that are on the 2540. Whenever operator intervention is needed, the new operator call light goes on, together with one or more of the lights on the backlighted panel. The lights identify the action to be taken rather than the error that occurred. For example, the check card light, instead of a validity check light, goes on when a card should be inspected for an invalid character or offpunching. There is also a replace 1 light that goes on together with other lights for the situation in which one card, instead of two,

should be placed back in the feed after a non-process-run-out. If the operator wants or needs even more explicit directions (for example, when more than one action can be taken, as for a format mode reset condition), he can find written directions in the error recovery procedure box, readily accessible under the card joggler plate.

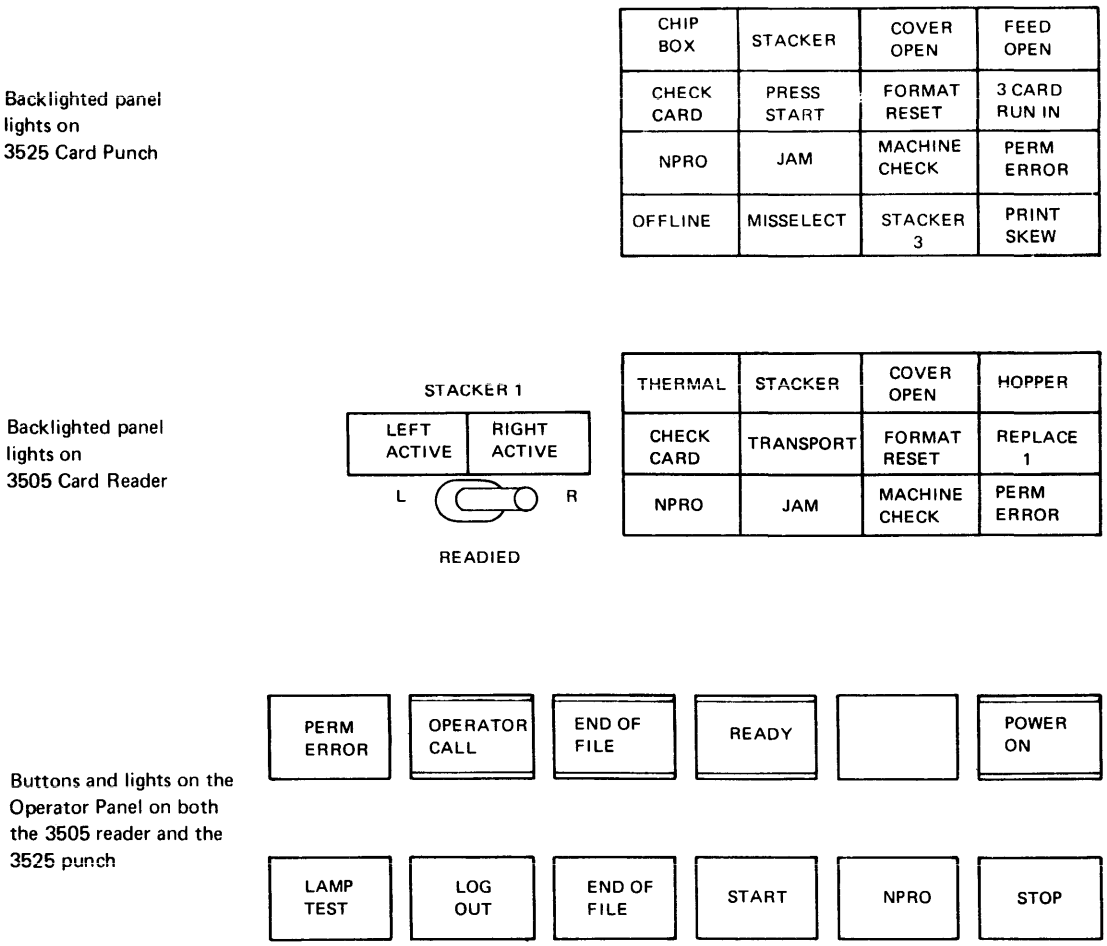


Figure 20.25.5. Contents of operator panels for 3505 reader and 3525 punch

MAINTENANCE

A new approach to maintenance of the 3505 and 3525 has also been taken that is designed to significantly reduce the time required to diagnose and repair failures.

When a unit error occurs on a 3505 or 3525, sense byte 2, and 3 if needed, is generated by the control unit for diagnostic purposes. Sense byte 2 is coded to uniquely identify errors. Byte 2 also serves as a pointer to the procedure to be taken as outlined in a new maintenance document, called the Graphic Integrated Manual, for customer engineers. This document contains information that currently is contained in several CE documents and it is designed to pinpoint failures (and the repair procedure required) without the use of an oscilloscope. Resolution of the error through the use of sense byte 2

and the new manual is designed to be such that nearly all errors can be diagnosed by the CE without the use of OLTEP and OLT's.

The error log number contained in sense byte 2 is available to the operator. It can be displayed on the operator panel and is printed on the console by OS and DOS. If this error number is reported to IBM when repair is requested, it may enable the customer engineer to determine whether or not the failure occurred because of a broken part. If a part is broken, the customer engineer can obtain a new part before going to the installation and, thereby, considerably reduce the time the reader or punch is unavailable.

As a further diagnostic aid, the control unit for the 3505 and the 3525 contains an error log storage area for each unit into which sense byte 2, and 3 if present, is automatically stored when an error occurs. The last 14 bytes of error log data for each unit are thus maintained for the CE to use during a maintenance period. The last error logged (sense byte 2) can be displayed on the operator panel by depressing the log out key. The entire log area of a 3505 or 3525 in offline status can be displayed on the CE panel in the 3505 without disturbing other system operations. (If the 3505 is placed offline, the 3525 can still continue normal operations, and vice versa.) The log area can also be read into processor storage for analysis and printing, using an OLT.

SUMMARY

The most significant advantages of the 3505 Card Reader and the 3525 Card Punch can be summarized as follows:

- Configuration flexibility - Choice of either of two read speeds combined with any one of three punch speeds
- New functions - Optical mark reading, card printing, and read column eliminate
- New RAS features - Automatic feed retry, friction feeding, optical reading, enhanced read checking, automatic punch retry, improved error recovery procedures, and more definitive hardware failure identification provided by the control unit
- Operator-oriented design - Alternate card stacking, easier jam removal, and explicitly defined intervention-required recovery procedures

Table 20.25.3. 3505 reader, 3525 punch, and 2540 reader/punch feature comparison

Feature	3505 Card Reader	2540 Card Read Punch (read features)	3525 Card Punch	2540 Card Read Punch (punch features)
Type of unit	Card reader and fully buffered control unit combined. Attaches to any channel. Operates independently of 3525, if the latter is attached.	Combined reader and punch unit. Requires 2821 control unit for attachment to any channel. Read and punch operations are independent and fully buffered.	Card punch unit attaches to channel via 3505. Additional control added to 3505 provides fully buffered operations independent from those of the reader.	See read features column.
Systems to which units attach	All System/370 models	All System/370 models and System/360 Models 25 and up	Same as 3505	Same as read features column
Models and speeds	B1 - 800-cpm read B2 - 1200-cpm read (combines with any 3525 model)	1000-cpm read (one model only)	P1 - 100-cpm punch P2 - 200-cpm punch P3 - 300-cpm punch (combines with any 3505 model)	300-cpm punch (one model only)
Card reading	Optical - serial by column	Eighty read brushes parallel by row	Card Read option required. Optical reading, parallel by row.	Punch Feed Read optional feature required. Read brushes for parallel reading by row.
Card feeding	Friction feeding with vacuum-assist.	Picker knives and multi-toothed clutch	Picker knives and multi-toothed clutch	Picker knives and multi-toothed clutch
Card feed retry after misfeed	Yes, up to three times	No	No	No
Read or punch checking	Each column read four times. Impulses checked for valid data or no-data indications.	Cards are read at two successive stations and hole counts are compared.	Punch checking done at punch station. Data from sensors attached to punches compared with data sent to be punched. Prepunched cards should not be used unless Card Read feature is present.	Punch checking done at 80-brush read station after punch station, as for read checking. Prepunched cards cannot be used unless Punch Feed Read feature is present.
Automatic punch retry after error	-	-	Standard hardware feature (for punch-only and punch/print operations).	Not provided by hardware. (Offered as an option by OS and DOS data management.)
File feed capacity	3000 cards	3100 cards	1200 cards	1350 cards
Stackers	One 3500-card-capacity logical stacker is standard (automatic alternate stacking in two 1750-card stackers). One additional 1750-card-capacity stacker is optional.	Two 1350-card-capacity dedicated read stackers are standard. Third 1350-card-capacity stacker is also standard and can be used by either read or punch side.	Two 1200-card-capacity stackers standard.	Two 1350-card-capacity dedicated punch stackers are standard. Third 1350-card-capacity stacker is also standard and can be used by either punch or read side.

Table 20.25.3 3505 reader, 3525 punch, and 2540 reader/punch feature comparison (continued)

Feature	3505 Card Reader	2540 Card Read Punch (read features)	3525 Card Punch	2540 Card Read Punch (punch features)
Read column eliminate	Optional (no-charge feature)	Not available	Standard with Card Read option.	Not available
Column binary operations	Standard (Card Image feature)	Optional (Column Binary feature)	Column binary punching is standard. Column binary reading is standard when Card Read is installed.	Column binary operations are provided when the Column Binary feature is installed for the read feed.
Optical Mark Reading	Optional (up to 40 vertically marked columns per card)	Not Available	Not available (with Card Read option)	Not available (with Punch Feed Read)
51-Column Interchangeable Read Feed	Not available	Optional	-	-
Card reading on punch unit	-	-	Optional (Card Read feature) and includes column binary reading. Read Column Eliminate is standard.	Optional (Punch Feed Read feature). Read Column Eliminate is not provided.
Card printing	-	-	Optional Two-line or Multi-line (up to 25) print features available	Not available
Error recovery	Control unit performs automatic feed retry. Control unit identifies error as permanent, retryable by ERP, or retryable by ERP after operator intervention. Operator informed of recovery action to be taken by lights on backlighted panel and writeup in error recovery procedures drawer. Permanent error key allows operator to terminate recovery action by ERP. Extended sense byte identifies unique errors and acts as interface to CE maintenance document.	Control unit does not perform recovery procedures. ERP inspects sense byte to determine recovery action to be taken. Lights on 2540 indicate error that occurred (no backlighted panel, permanent error key, or error recovery procedures drawer). Only one sense byte presented.	Control unit performs automatic punch retry. Other procedures are same as those for 3505. Same ERP used for reader and punch.	Same as read features column. (Punch retry is programmed and optional.)

THIS PAGE INTENTIONALLY LEFT BLANK

SECTION 30: PROGRAMMING SYSTEMS SUPPORT

30:05 TRENDS IN DATA PROCESSING AND PROGRAMMING SYSTEMS

The Model 135 and its programming systems support have been designed to operate in the data processing environment that has been emerging since the introduction of System/360.

Significant trends are the following:

- Growth toward more multiprogramming to improve system throughput. Multiprogramming also permits the user to install new applications, such as small teleprocessing inquiry or graphics applications, that otherwise would not justify a dedicated system. Multiprogramming support also has encouraged the growth of new computer environments, as indicated by the items that follow.

- Growth of integrated emulation, that is, concurrent native and emulation mode processing on one system. The execution of emulators under operating system control improves system throughput because emulators can use control program facilities (stacked job execution, data management functions, etc.) and because native mode and emulator jobs can be scheduled to operate concurrently to utilize available system resources more efficiently. The use of integrated emulators eliminates most reprogramming and eases transition from one system to another, permitting the user to expend his efforts extending and adding applications.

- Greater use of high-level languages, such as COBOL, FORTRAN, and PL/I, for applications programming. The cost of programming has been increasing, while the cost of computing hardware has been decreasing. More productive use of programmers can be achieved by the use of high-level languages. Improvements to compile times and to the size and execution speed of code produced by high-level language translators have been made and continue to be made. The support of many more functions within high-level languages has also increased their use, and the growth of interactive computing has stimulated the addition of even more facilities.

- Growth of teleprocessing applications, such as remote inquiry, message switching, data collection, and management information systems. The ability of System/360 and System/370 to handle teleprocessing and batch processing in one system eliminates the necessity of dedicated, special purpose systems.

- Growth of remote computing activities, such as remote job entry and interactive computing (or time sharing), in both a nondedicated and a dedicated environment. Remote computing offers (1) fast turnaround for batch work submitted from remote locations, (2) remote user access to the large computing facilities and data base available at the central installation, and (3) interactive problem solving on a regular or a nonscheduled basis for personnel in locations remote from the central computer. In-house interactive computing is growing also as users attempt to use programmer time more efficiently.

- Growth toward large, online data base systems. The growth in the marketplace of remote computing, time-sharing, and real-time applications necessitates the instant availability of more and more data. High-capacity, fast, low-cost, reliable direct access devices supported by appropriate data organizations, access

techniques, and security measures will be required for this type of computing environment.

IBM programming systems support by DOS and OS of these trends in data processing has been growing and continues to expand. The System/370 Model 135 offers hardware, I/O devices, and performance capability required by the expanding computing environment.

30:10 DOS SUPPORT

DOS will be modified and extended in future releases so that it supports certain Model 135 hardware features and I/O devices. Appropriate alteration of the DOS supervisor generated for a Model 135 will allow it to accommodate the fixed area of lower processor storage in the Model 135. DOS for the Model 135 includes currently announced DOS facilities and additional support to handle certain Model 135 hardware features and I/O devices.

DOS support of Model 135 features will be provided as follows (programming systems support of RAS features is discussed in Section 50):

New instructions. Assembler D (14K variant) will include mnemonics for all the new System/370 instructions so that they can be used in user-written Assembler Language programs. The DOS high-level language translators currently offered will not generate the six new general purpose instructions.

Extended precision floating point. Mnemonics for extended precision instructions and data formats will be added to Assembler D (14K variant). A simulator for extended precision operations, such as that provided by OS, is not provided by DOS. The DOS high-level language translators currently offered do not support extended precision.

Interval timer. The timer will be supported for the same functions as it is currently, for time of day and time intervals.

Time of day clock. This clock is not supported for time of day values.

Byte boundary alignment. The programmer has the ability to byte-align binary and floating-point data in Assembler Language programs, in ANS COBOL programs (by omitting the SYNCHRONIZED clause), and in PL/I programs (by specifying the UNALIGNED attribute). However, COBOL and PL/I still align unaligned fixed- and floating-point data prior to its use.

1401/1440/1460 Compatibility feature. A 1401/1440/1460 integrated emulator program will be provided. (See Section 40 for a complete discussion of the emulator program.)

New console devices. The 3210 Model 1 and the 3215 Console Printer-Keyboards are supported as the DOS console device.

Channels. The byte multiplexer channel, with up to 64 subchannels, and selector channels are supported. Block multiplexer mode is not supported.

Integrated File Adapter. The IFA and 2319 DASF do not require any special programming support and will be supported in the same manner as 2314 disk storage.

Integrated Communications Adapter. BTAM and QTAM will support the same functions for terminals connected via the ICA as are currently provided for these terminals connected via a 2701. The Read Interrupt

and Write Interrupt optional features are not supported.

The 3330 facility. The 3330 will be supported for the same system control and system service functions as is 2314 disk storage. That is, the 3330 will be supported as a system residence, a system input, and a system output device and by the linkage editor, librarian, and special utilities programs. Data management support of the 3330 as an I/O device, functionally identical to that available for the 2314, is provided by Assembler D (14K Variant).

The rotational position sensing facility is not supported. That is, SET SECTOR and READ SECTOR commands will not be included in 3330 channel programs. Block multiplexing is not supported and a channel with 3330 facilities connected operates in selector mode. Therefore, only one data transfer channel program can be in operation on the channel at a time. However, stand-alone seek operations can be overlapped with one another and with one data transfer operation, as for a 2314, when the seek overlap option is present in the DOS supervisor.

3211 Printer. This printer will be supported in the same manner as is the 1403 Printer, including support by DOS POWER II. New parameters will be added to some of the existing printer Assembler Language macros to support new features of the 3211. The 18 additional print positions are supported only by the Assembler Language. Forms control buffer and universal character buffer loading for the 3211 will be handled in the same way. The user must execute an IBM-supplied buffer load utility program (SYSBUFLD) as a job step in order to load the FCB and/or the UCB. No provision has been made for loading the FCB or UCB during execution of a job step. User-defined UCB images must be loaded from a core image library. FCB images can be loaded from cards or a core image library.

If a command retry indication is present, the 3211 error recovery routine supports retry of an operation that failed. This option must be requested by the user in the DTF for the 3211 Printer.

ASCII mode tapes. The capability of processing ASCII mode tapes is provided by the following program products:

- ANS Full COBOL Compiler V3
- ANS Full COBOL Object-Time Library
- ANS Subset COBOL Compiler and Library
- PL/I Optimizing Compiler
- PL/I Transient Library
- FORTRAN IV Library
- ASCII Magnetic Tape Utilities
- Tape and Disk Sort/Merge
- RPG II

An initialize tape utility is provided that can be used to write an ASCII mode standard volume label on tapes. ASCII mode tapes are also supported by data management (the sequential access method) and thus they can be processed by user-written Assembler Language programs.

The 3803/3420 Magnetic Tape Subsystem. This tape subsystem will be supported as an I/O device for the same functions as 2400-series tape units. This includes support of tape switching (a maximum of two control units), Seven-Track, and Dual Density features. (Note that 200-BPI-density tapes are not supported because the Seven-Track feature includes only 556- and 800-BPI densities.) The Two-Channel Switch is supported for alternate tape switching. The RESERVE and RELEASE commands are not supported.

The 3505 Card Reader. This reader will be supported at the GET/PUT macro level by Assemblers D and F for basic read functions, stacker selection, and column binary mode. The 3505 will also be supported by all DOS high-level languages to the same extent that support for card readers is provided currently. Support of the OMR and RCE features is provided in the Assembler Language only. However, RPG II provides a facility by which the user can include Assembler Language routines to support the OMR, RCE, or Card Image feature in an RPG II program.

The 3505 is also supported as a SYSRDR, SYSIPT, or SYSIN device, and by DOS POWER II for basic read functions only. If OMR or RCE is to be used by an Assembler Language program, the 3505 cannot also be used as the SYSRDR, SYSIPT, or SYSIN device.

Support of 3505 features by the Assembler is as follows:

- Stacker selection into stackers 1 and 2 is provided. If a request for stacker 3 is received, the card is placed in stacker 2 by the 3505. If a request for stacker 2 is received and the optional Selective Stacker feature is not installed, the 3505 places the card in stacker 1. No indication of either action is given to the user program.

- Card Image support is provided as follows. Either EBCDIC or column binary read mode is established at OPEN time, depending on the user specification, and this mode remains in effect until the file is closed. Thus input decks that contain cards with EBCDIC and column binary punching in the same card or a mixture of cards punched in all one mode or the other must be handled by the user with the EXCP macro.

- Either OMR or RCE support can be requested only in the DTFCD macro. DTFDI can be used to support the 3505 only for basic read functions. The OMR or RCE format to be used must be supplied in the first card of the input deck. Once the file has been opened and the format has been established, the format remains in effect until the file is closed or until unit exception (end of file) on the 3505 is accepted by the system.

Utility support of the 3505 and 3525 is provided by the copy disk-to-card and restore card-to-disk program contained in the Group 1 Utilities (Unit Record and Disk).

The 3525 Card Punch. This punch and all its special features will be supported at the GET/PUT level by the Assembler Language. The basic punch function of the 3525 will also be supported by DOS POWER II. The 3525 is supported as a SYSPCH unit and, if the read feature is present, as a SYSIN device. Except for RPG II, the DOS high-level languages support punch or read operations on the 3525 only as currently provided, and thus RCE, card printing, and two or more combined operations on each card (reading, punching, and printing) are not supported. RPG II supports the Card Print feature and two or more combined operations (reading, punching, printing) on each card. RPG II does not support RCE or Card Image directly in the language but supplies a facility to permit the user to include Assembler Language routines to support these features.

The 3525 is supported by the Assembler Language and RPG II for the following functions:

- Punching only - Stacker selection is supported.

- Reading only - Includes support of Card Image, RCE, and stacker selection as described for the 3505. (RPG II support of Card Image and RCE requires inclusion of a user routine.)

- Printing only - Programming is similar to that for a printer. Functionally, a card is considered to be a page with 25 possible print lines. Varying the stacker selected by programming is not supported. All cards will be directed to stacker 1.

- Interpreting only - This is actually a punch and print combination in which the data punched is also printed on line 1 (first 64 characters) and line 3 (remaining 16 characters) as shown previously in Figure 20.25.4.

- Any combination of reading, punching, and printing - The concept of associated files is used for handling two or more operations on the same card, instead of the combined file technique currently supported for read/punch operations on a 2540 with Punch Feed Read. A separate file (read, punch, or print) is defined by the user for each function to be performed. These files are allocated to the same 3525 via ASSGN statements and their DTF's contain a new parameter to indicate that they are associated files. When macros are issued, a certain sequence must be maintained.

Utility support for the 3525 is described under "The 3505 Card Reader".

30:15 OS MFT SUPPORT

OS MFT will be modified and extended in future releases so that it supports Model 135 hardware and I/O devices. (PCP and MVT are not being extended to support model-dependent features of the Model 135.) Appropriate alteration of the resident portion of a control program (nucleus) generated for a Model 135 will accommodate the fixed area of lower processor storage in the Model 135. OS for the Model 135 includes currently announced OS MFT facilities and contains additional support to handle new Model 135 hardware features and I/O devices. Emphasis also has been placed on extending error recovery, recording, diagnostic, and repair procedures.

OS MFT support of Model 135 features will be provided as follows (programming systems support of RAS features is discussed in Section 50):

New instructions. The Assembler F (Type I) and Assembler H (program product) language translators will include mnemonics for the general purpose and other new instructions for the Model 135 so that these instructions can be used in user-written Assembler Language programs. The currently offered OS high-level language translators will not generate the six new general purpose instructions.

Extended precision floating point. Assemblers F and H will include support of the extended precision floating-point data format and instructions. In addition, extended precision will be supported by the FORTRAN H, PL/I Optimizing Compiler, and PL/I Checkout Compiler program products.

The implementation of extended precision support by FORTRAN H and PL/I is such that the language translators and the processing programs they generate to include extended precision operations can operate on a System/370 or a System/360 with or without extended precision hardware. The language translator contains extended precision instructions and generates them for processing programs that use extended precision data. A program check interrupt occurs if an extended precision instruction is executed and the feature is not present in the system. This interrupt causes the processing program to call in a subroutine (the extended precision floating-point simulator) to handle extended precision operations. (The extended precision simulator, which simulates all the operations provided by

extended precision instructions and extended precision divide as well, is automatically included in all OS operating systems at system generation time.) Extended precision divide is always simulated, since the extended precision hardware feature does not include such an instruction.

Interval timer. The interval timer will be supported for the same functions as it is currently except for time of day values.

Time of day clock. This clock will be supported for time of day requests made by system and user tasks via the TIME macro. At IPL, the operator will have the option of validating the clock time and correcting an invalid value.

Byte boundary alignment. The programmer has the ability to byte-align binary and floating-point data in Assembler Language programs, in ANS COBOL programs (by omitting the SYNCHRONIZED clause), in PL/I programs (by specifying the UNALIGNED attribute), and in FORTRAN programs. However, OS still expects parameters passed to it to be properly aligned and the high-level language translators still align unaligned fixed- and floating-point data before it is used.

1401/1440/1460 Compatibility feature. A 1401/1440/1460 integrated emulator program will be provided. (Emulator programs are discussed in Section 40.)

OS/DOS Compatibility feature. An OS DOS emulator program will be provided to support emulation of a DOS system. (This emulator is discussed in Section 40.)

New console devices. The 3210 Model 1 and 3215 Console Printer-Keyboards are supported as the primary operating system console device.

Channels. The byte multiplexer channel with up to 64 subchannels is supported. Selector and block multiplexer mode are supported also. During IPL, channel mode for all installed block multiplexer channels is established via a control register channel mode bit setting based on system generation channel definitions. (The channel mode bit is discussed in Section 10:20.) The operator does not have the option of changing this mode at IPL, nor does the control program change the mode setting during system operation.

Integrated File Adapter. The IFA and 2319 DASF do not require any special programming and will be supported in the same manner as 2314 disk storage.

Integrated Communications Adapter. BTAM, QTAM, and TCAM will support the same functions for terminals attached via the ICA as are currently provided for these terminals attached via a 2701. The Read Interrupt, Write Interrupt, and Unit-Exception Suppression optional features are not supported.

The 3330 facility. The 3330 facility will be supported as an I/O device for most of the same functions as is the 2314 facility and by ASP and HASP II. The error recovery routine provided for the 3330 will include support of the new hardware error correction features.

RPS will be supported as follows.

- The stand-alone seek issued within the I/O supervisor (IOS) will be eliminated for RPS devices (3330 facilities). IOS will continue to issue stand-alone seeks for direct access devices without RPS. IOS also will be capable of recognizing the channel available interrupt.

- Access method support of RPS commands (SET SECTOR and READ SECTOR) will be provided by QSAM, BSAM, BPAM, BDAM, and ISAM. The sector number will be calculated for fixed-length records when possible or obtained by use of the READ SECTOR command (for example, during sequential processing of variable-length records). Specifically, RPS will supported by:

THIS PAGE INTENTIONALLY LEFT BLANK

QSAM and BSAM for all record formats and functions provided for the 3330 facility except the undefined track overflow record format

BPAM for processing directory and member records wherever possible. (Directory entries will not be modified to include sector values.)

BDAM for direct retrieval and update of fixed-length standard and VBS format records without key, and for write verification of all BDAM record formats

ISAM for:

1. All operations involved in data set creation (fixed- and variable-length records) using QISAM load mode. (Index entries will not be modified to include sector values.)
2. Sequential retrieval of fixed- and variable-length prime and overflow records using QISAM scan mode (all prime records except the first on the track)
3. Addition of new records to the prime and overflow areas, including the searching of overflow chains, using BISAM
4. All validity-checking operations (data and index records)
5. All updating operations (data and index records)
6. Operations that require orientation to the beginning of the track before processing, such as index searching, BISAM direct retrievals, reading the first data record during sequential operations, etc. (A sector value of zero is used.)

- End-of-volume (EOV) routines will support concatenation of data sets residing on RPS and on non-RPS devices. The control program will ensure that an RPS access method is used for drives with the feature and that a non-RPS method is used for drives without the feature.

- Any system utility, data set utility, or IBM-supplied processing program (such as a language translator) that uses the sequential access methods will support RPS. In addition, IEHMOVE, IEBCOPY, and the initialize/dump/restore utility (IEHDASDR) will include RPS support for 3330 facilities.

- The Sort/Merge (program product) supports RPS for 3330 intermediate work devices.

- Where appropriate, RPS commands for access to SYSRES data sets will be supported by:

  Data set catalog routines
  Direct access device space management (DADSM routines) for DSCB processing
  STOW, BLDL, and FIND processing of program library (PDS) directory entries
  OPEN/CLOSE/EOV processing of JFCB's in the job queue
  Routines that access the job queue

Note that RPS command support is not provided for:

Program fetch
Access to TSO swap data sets
Telecommunications access method (TCAM) message queue processing
The stand-alone disk initialization and alternate track assignment routines (DASDI and IEHATLAS)

3211 Printer with tapeless carriage. The 3211 Printer, with or without the 18 additional print positions, will be supported by QSAM and BSAM for exactly the same functions as is the 1403 Printer and by ASP and HASP II. In addition, the control program will handle loading of the forms control buffer (FCB) with carriage control images. This support is similar to that provided for Universal Character Buffer (UCB) loading.

The user can define one or more default FCB images at system generation time. Two IBM-supplied default images are included automatically. All other FCB images to be used must be defined by the user and placed in SYS1.SVCLIB, as is the case with UCB images. User-supplied default images must be identified as defaults. The FCB image to be used by a processing program can be specified in the 3211 Printer DD statement included for the job step and will be loaded into the FCB by the control program.

The FCB image currently loaded can also be changed by the programmer during execution of the processing program by use of an Assembler Language macro. If the DD statement does not specify an FCB image and the image currently loaded is not one of the defaults specified at system generation, the operator is requested to specify the FCB image to be used.

The FCB image (and the UCS character image) to be loaded for a 3211 Printer used by an output writer can be indicated in the output writer procedure or in the START output writer command issued by the operator. FCB and UCB images can also be specified in the SYSOUT DD statement for a data set that is to be printed by the output writer, and they will be loaded into the 3811 control unit prior to the printing of the data set.

Any time the FCB parameter is used, as described above, the user can specify that operator verification of forms alignment is to be requested by the control program via a console message when the buffer is loaded. The operator must respond to this message.

The 3211 error recovery routine will retry a print operation after a parity check occurs in the UCB or print line buffer if QSAM is used and three I/O buffers are provided for the printer data set. When the operation is retried, the 3811 control unit ensures that only the print positions that did not print correctly the first time are reprinted.

ASCII mode tapes. The capability of processing ASCII mode tapes is provided by the following program products:

- ANS Full COBOL Compiler V3
- PL/I Optimizing Compiler
- PL/I Checkout Compiler
- PL/I Transient Library
- Sort/Merge
- FORTRAN IV Library (Mod I)
- FORTRAN IV Library (Mod II)
- Data Set Utilities - a set of four ASCII utilities that includes printing, punching, and comparison of ASCII mode tapes in addition to translation to and from ASCII mode
- TSO Data Utilities

ASCII mode tapes are also supported by data management (QSAM and BSAM) and thus they can be processed by user-written Assembler Language programs.

The 3803/3420 Magnetic Tape Subsystem. This tape subsystem will be supported as an I/O device for the same functions as 2400-series

tape units. This includes support of tape switching, Seven-Track, and Dual Density features. (Note that 200-BPI-density tapes are not supported, because the Seven-Track feature includes only 556- and 800-BPI densities.) The Two-Channel Switch is supported for alternate path switching. The RESERVE and RELEASE commands are not supported.

The 3505 Card Reader. This reader will be supported by QSAM and BSAM for the same basic input functions as are provided for the 2540 and 2501 readers, including support as a SYSIN device. Features will be supported by the access methods as follows:

- Stacker selection into stackers 1 and 2 is provided, as for current readers. If a stacker select request for stacker 3 is received, the 3505 will place the card in stacker 2 without giving any indication of this action to the user program. If a request for stacker 2 is received and the optional Selective Stacker feature is not installed, the 3505 will direct the card to stacker 1 without any indication to the program.

- Card Image support is identical to that currently provided. Either EBCDIC or column binary read mode is established at OPEN, depending on the user specification, and this mode remains in effect until the data set is closed or end of file occurs on the reader. Thus input decks that contain cards with EBCDIC and column binary punching in the same card, or a mixture of cards punched in all one mode or the other, must be handled by the user with the EXCP macro.

- OMR and RCE are supported but not concurrently for the same data set. These options are requested via the DCB MODE parameter, either in the program DCB or in the DD statement DCB parameter. The format to be used is supplied by the user in the first card of the input deck. Once the data set is opened and the format has been established, the format remains in effect until the data set is closed. A data set to be read in OMR or RCE format mode cannot be placed in the input stream (in the SYSIN device). It must be read directly from the 3505 by the user program. A program that is to use one of these features must specifically request allocation of a 3505 with the feature installed, via the UNIT parameter, as the scheduler will not allocate a 3505 by feature.

The OS high-level languages (COBOL, PL/I, FORTRAN, and RPG) also support the 3505, including OMR and RCE. Support for these new features can be requested in job control statements and is, therefore, transparent to the high-level languages. The Card Image and Stacker Select features of the 3505 are supported by these languages to the same degree as they are for other readers. Any OS utility that uses QSAM or BSAM also supports the 3505 and its features. ASP and HASP II will support the 3505 for the same functions provided currently.

The 3525 Card Punch. This punch will be supported by QSAM and BSAM for the same basic punch and stacker selection functions as are provided for the 2540 punch. It will also be supported as a SYSOUT device and, if the read feature is present, as a SYSIN device. (RCE cannot be used in the latter situation.) All optional 3525 features will also be supported by the Assembler Language and by all OS high-level languages except ALGOL. The 3525 can be used to perform the following functions:

- Punching only - Either EBCDIC or binary data can be punched and stacker selection is supported.

- Reading only - Includes support of Card Image, RCE, and stacker selection as described for the 3505 reader

- Printing only - Programming is similar to that for a printer. Functionally, a card is considered to be a page with 25 possible print lines. Varying the stacker selected by programming is not supported; however, all cards can be directed either to stacker 1 or to stacker 2.

- Interpreting only - This is actually a punch and print combination in which the data punched is also printed on line 1 (first 64 characters) and line 3 (remaining 16 characters) as shown previously in Figure 20.25.4.

- Any combination of reading, punching, and printing - The concept of associated data sets is used for handling two or more operations on the same card. (QSAM and BSAM do not currently support combined read/punch operations on a 2540 with Punch Feed Read.) A separate data set is defined by the user for each function (read, print, or punch) to be performed. These data sets are allocated to the same 3525 using the DD job control statement (AFF parameter), and when macros are issued, a certain sequence must be maintained. The advantage of this approach is that it is transparent to all high-level languages and it provides device independence. For example, a program written in a device-independent manner designed to read, punch, and print using the 3525, can be executed using a tape unit, a direct access device, and a printer for the three respective operations, if necessary, merely by changing job control statements.

OS utilities that use QSAM or BSAM will support the 3525 and its optional features as described above. ASP and HASP II will support the 3525 for the same funcions currently provided.

40:05 DOS 1401/1440/1460 EMULATOR PROGRAM

GENERAL OPERATION

The Model 135 continues the advantages of integrated emulation available to Model 25 and 30 DOS CS/30 users.

A 1401/1440/1460 Emulator program is provided, which runs as a problem program under DOS on a Model 135 equipped with the 1401/1440/1460 Compatibility feature. This no-charge compatibility feature is also used by the Model 135 OS 1401/1440/1460 emulator.

The DOS 1401/1440/1460 emulator for the Model 135 provides the same functions as the DOS 1401/1440/1460 emulator for System/370 Models 145 and 155. The 1401/1440/1460 compatibility features on Models 135 and 145 are identical but differ slightly in implementation (for SAR and SBR instructions) from the 1400 compatibility feature on the Model 155. Therefore, a 1400 emulator program generated for the Model 135 will run on the Model 145 but not on the Model 155. However, a 1401/1440/1460 emulator generated for the Model 155 will run on all three models.

The emulator can be used in a batch-only system environment or can operate in the background and batched foreground partitions of a multiprogramming system. Therefore, more than one 1401/1440/1460 Emulator program can execute concurrently with each other and with System/370 programs. Additionally, emulated jobs and DOS jobs can be intermixed in a single job stream.

The Model 135 DOS integrated emulator consists of the compatibility feature, simulation routines, and DOS data management routines. It offers Model 135 users the following advantages:

- System resources are more fully utilized.
- Emulators can run concurrently in all three partitions of a multiprogramming system. They are relocatable and can be link-edited to run in any partition.
- 1401/1440/1460 Emulator programs and DOS programs can be executed concurrently and intermixed in a single job stream.
- DOS supervisor and data management services are available to the user. This provides job control facilities, standard disk and tape label processing, and common data formats for emulator files and DOS files.
- 1400 unit record input/output operations are device independent and can be emulated on Model 135 unit record devices, magnetic tape units, or direct access storage devices.

For the Model 25 or 30 DOS CS/30 user, the Model 135 DOS 1401/1440/1460 emulator continues the advantages of integrated emulation and provides additional advantages, such as:

- Emulators operating concurrently in all three partitions
- Support of CS/30 disk and tape files and emulator control statements
- System/370 data formats for emulator tape and disk files
- Elimination of the preformatting of disk packs used for output
- Added emulator control available to user at execution time
- DOS data management facilities and standard disk and tape label processing

Emulator Program Generation and Execution

The Model 135 DOS 1400 emulator is distributed with DOS releases. An emulator is assembled by the use of macro instructions. The macro instructions describe the 1400 CPU, input/output devices, special features, data files, emulator buffers, and the desired user options. When assembled, the macros provide an object module and linkage to preassembled modules stored in the system relocatable library. The preassembled modules are combined with the emulator object module by the linkage editor for cataloging in a core image library. Any number of emulators can be assembled and cataloged in a core image library to run in any partition.

The emulator program generated will emulate, without change, 1400 programs written in accordance with IBM 1400 Principles of Operation manuals, subject to the following conditions:

- 1400 programs that purposely depend on the absence of a 1400 feature or on error conditions may not execute properly.
- Programs with undetected programming errors and those that depend on timing of 1400 I/O operations yield unpredictable results.

An emulator program is handled by DOS in the same manner as any problem program. When using the 1401/1440/1460 emulator, 1400 programs may be cataloged to, and fetched from, a core image library for execution or loaded from a card, tape, or direct access storage device. Standard DOS job control statements are used to prepare the system for an emulator job. The EXEC job control statement causes the specified emulator program to be loaded and control is passed to the emulator program. Emulator control statements are read by the emulator from the logical unit selected at generation time.

Emulation with the 1401/1440/1460 emulator consists of three main steps:

1. Initialization. Emulator control statements supplied by the user are read and interpreted. This information overrides, for the execution of the emulator, information specified at emulator generation.
2. Loading or fetching. The 1400 program is loaded from a card reader, magnetic tape unit, or direct access storage device. A 1400 program can also be fetched from a core image library if it has been cataloged.
3. Execution or precataloging. When loaded, the 1400 program is executed. The 1400 instructions are fetched, interpreted, and executed by the emulator until an end-of-job condition is recognized. The 1400 program can be either executed or converted into a DOS object module (precataloged). This module can be subsequently link-edited and cataloged in a core image library.

Input/output errors are processed by DOS device error recovery procedures. Input/output errors that cannot be corrected, such as permanent input/output errors and wrong-length records, are passed to the 1400 program.

Console simulation and operator communication with the emulator program are provided by the exchange of emulator commands and messages between the operator and the emulator program. The emulator provides messages to inform the operator of errors or other conditions that require his attention or a response. Emulator commands can be entered from the console printer-keyboard and are handled in the same way as operator communications are currently handled by DOS.

Tape and Disk Emulation

The user has the option of processing tape files in 1400 format or in spanned variable-length record format. Two tape formatting programs--the Tape Preprocessor and Tape Postprocessor--are available to convert tape files from 1400 format to spanned record format, and vice versa; 1400 tapes containing records larger than 32K must be converted to spanned record format prior to emulation. Mixed-density tapes are not supported by the emulator or the tape formatting programs.

The emulator accepts as input two tape file formats:

1. 1400 format, which is produced by a 1400 system, a stand-alone emulator, CS/30, the Tape Postprocessor formatting program, or a Model 135 1400 emulator

2. Spanned variable-length record format, which is produced by the Tape Preprocessor formatting program or a 1400 emulator

Either format can be produced as output by the emulator.

Processing tape files in spanned variable-length record format provides several advantages:

- Blocking short records reduces the time for emulating I/O operations.

- The Tape Preprocessor or the Tape Postprocessor program can be run concurrently with emulator programs in a multiprogramming system.

- Files in spanned variable-length record format can be used by other Model 135 programs if the programs provide for handling the 1400 label records and 1400 tapemark records.

- The Tape Postprocessor program can be used to convert a file in spanned variable-length record format back to 1400 format for use on a 1400 system.

Tape files in spanned record format have standard DOS labels; 1400 labels are treated as data records, since they are processed by the 1400 program. The 1400 tapemarks appear as special data records and are recognized by the 1400 emulator.

The character codes supported by the 1400 emulator for magnetic tape data are:

- BCD representation in even and odd parity for seven-track tape (data translator on) in 1400 format

- BCDIC-8 representation for nine-track tapes in either 1400 or spanned record format, and for seven-track tapes (data converter

on) in spanned record format. This character code, which is the eight-bit representation of BCD, is used to simulate parity. In normal mode, bit 1 is set to one for even parity, to zero for odd parity. In alternate mode, bit 1 is always set to one and no distinction is made between even and odd parity.

Two tape formatting programs, a preprocessor and a postprocessor, are available for converting tape files. They are distributed to run as problem programs under DOS control and operate only in the background partition. The preprocessor requires a partition size of 7300 bytes plus I/O areas and the postprocessor requires 6300 bytes plus I/O areas.

The Tape Preprocessor program converts seven-track or nine-track tapes in 1400 format to seven-track (data converter on) or nine-track tapes in spanned variable-length record format with standard DOS labels. The Tape Postprocessor converts seven-track or nine-track tapes in spanned record format to seven-track or nine-track tapes in 1400 format.

Disk files in 1400 format, which are created on a 1400 system or under stand-alone emulation, must be converted to a standard fixed-length record format on 2311 or 2314 disk packs before emulation. Disk files created under CS/30 can be processed by the 1401/1440/1460 emulator if the CS option is specified at emulator generation.

Existing 1400 utilities and the DOS Clear Disk utility program are used to convert disk files in 1400 format, and CS/30 files if desired, to the standard record format.

Each Model 135 disk record represents one 1400 disk track. Each Model 135 disk record is a fixed-length record, its length being a function of the emulated 1400 device and mode rather than the amount of 1400 data on each track. A 1400 disk file can occupy one or more extents on Model 135 disk packs but only one extent per pack. Extents must be allocated complete cylinders. When a file requires more than one Model 135 disk pack, the packs must be the same type. Two different 1400 files can be placed on the same disk pack but this arrangement may increase seek time if both files are processed at the same time.

Character codes supported by the 1400 emulator for disk files are:

- EBCDIC representation for disk operations in move mode
- BCDIC-8 representation for disk operations in load mode. (Data written in load mode must be converted to EBCDIC if it is to be used by programs other than the emulators.)

To convert disk files in 1400 format, or CS/30 disks if desired, to a standard format on a Model 135 disk, the user must dump and restore the data as follows:

1. Dump the disk device, using a 1400 disk-to-tape or disk-to-card utility program. When converting files on 1301, 1311, or 1405 disk devices that were created on a 1400 system, the utility is executed on the system used to create the file. When converting files on 2311 or 2314 disks that were created under stand-alone emulation or CS/30, the utility is executed on a System/360 under control of the emulator used to create the disk file.

2. Use the DOS Clear Disk utility program to format previously initialized 2311 or 2314 disk pack(s) for the data.

3. Restore the data to a formatted 2311 or 2314 disk pack, using a 1400 tape-to-disk or card-to-disk utility program under control of a Model 135 1401/1440/1460 emulator.

Emulator performance will vary depending on user options, such as number and size of buffers, the instruction mix of the 1401/1440/1460 programs, the format of tape files, and the priority of the partition in which the emulator is running.

Emulator performance is improved by:

1. Using double buffers and spanned record format for tape files in lieu of single or shared buffers and 1400 record format. (A shared buffer can be used by more than one I/O device.)

2. Using single buffers rather than shared buffers for disk files

3. Specifying device independence for emulating unit record operations on a magnetic tape or direct access storage device

4. Using a card reader that is not equipped with the 51-Column Interchangeable Read Feed and Column Binary features, and not using the Select Stacker instruction

SUPPORT OF 1401/1440/1460 FEATURES

The size of the partition required for emulation depends on the 1400 system being emulated, including standard and special features, input/output devices, buffers, etc. The processor storage required for the 1401/1440/1460 emulator is equal to the combined sizes of:

- Simulated 1401/1440/1460 storage. Each position of 1400 storage is simulated in one byte of Model 135 storage (for example, 8000 positions = 8000 bytes).

- Emulator routines required to emulate the 1401/1440/1460 system instructions, features, and I/O operations

- Tape, disk, and unit record buffers. The number and size of tape and disk buffers are specified by the user.

Estimated minimum 1401/1440/1460 emulator processor storage requirements for emulation of a 1400 system with unit record operations only, unit record/tape operations, or unit record/tape/disk operations are shown below.

Emulated Operations	DOS Partition Size (bytes)
1401/1440/1460 unit record	17K + 1401/1440/1460 storage size + buffers
1401/1440/1460 unit record and 6 tapes	23K + 1401/1440/1460 storage size + buffers
1401/1440/1460 unit record, 6 tapes, 4 disks	27K + 1401/1440/1460 storage size + buffers

The 1400 CPU features and 1400 I/O devices and special features supported and the Model 135 devices used for 1401/1440/1460 emulation are given in Tables 40.05.1 and 40.05.2. Table 40.05.3 lists the 1400 I/O devices that are not supported.

Table 40.05.1. 1401/1440/1460 I/O devices and features supported by the DOS 1401/1440/1460 Emulator program and corresponding Model 135 devices

1401/1440/1460 Device and Features	Corresponding Model 135 Device
1402, 1442, 1444 Card Read Punch with stacker selection The following are supported: • Column Binary or Card Image • 51-Column Interchangeable Read Feed • Punch Feed Read • Punch Column Skip • Binary Transfer • Processing Overlap • Read Punch Release (as a no-op) Not supported: • Multiple card reader/punch operations in one program	1442, 2520, 2540 Card Read Punch, 2501 Card Reader, 3505 Card Reader, 3525 Card Punch Note: Card reader and card punch operations may be emulated using a magnetic tape or direct access storage device.
1403, 1404, 1443 Printer The following are supported: • Numerical Print • Processing Overlap • Space Suppression Not supported: • Selective Tape Listing • Multiple printer operations • Cut-card operations and read compare	1403, 1443, 3211 Printer Note: Printer operations may be emulated using a magnetic tape or direct access storage device.
1407, 1447 consoles	3210 Model 1 or 3215 console
729, 7330, 7335 Magnetic Tape Unit The following are supported: • Binary tape instructions • Processing Overlap Not supported: • Compressed tape	2400- and 3400-series magnetic tape units • Seven-Track feature is required if processing seven-track tapes
1301, 1311, 1405 Disk Storage The following are supported: • Direct Seek • Scan Disk • Track Record • Additional access arm (1405) Note: A 1405 cannot be emulated in combination with a 1301 or 1311	2311, 2314-type, 3330 direct access devices

Table 40.05.2. 1401/1440/1460 CPU features supported by the DOS 1401/1440/1460 Emulator program

Storage from 1400 to 16,000 positions. The 1401 Model G is not emulated.	Multiply-Divide
Expanded Print Edit	Sense Switches
Inverted Print Edit	Advanced Programming
High-Low-Equal Compare	Indexing and Store Address Register
Move Binary Code and Decode	Bit Test

Note: Translate feature is not supported.

Table 40.05.3. 1401/1440/1460 devices not supported by the DOS 1401/1440/1460 Emulator program

1445 Printer	1404 Printer in cut-card mode
Paper tape readers	7340 Hypertape Drive
Paper tape punches	Teleprocessing devices
Magnetic character readers	Audio response units
Optical character readers	

40:10 OS 1401/1440/1460 EMULATOR PROGRAM

GENERAL OPERATION

One of the significant new features of the Model 135 for OS PCP users is support of integrated 1400-series emulation. A 1401/1440/1460 emulator program that operates under OS MFT control on the Model 135 is provided. As discussed for DOS at the beginning of Section 40:05, an OS 1401/1440/1460 emulator program generated for the Model 135 will run on a Model 145 but not on a Model 155 and an OS 1401/1440/1460 emulator generated for a Model 155 will run on all three models.

An emulator program requires the presence of the 1401/1440/1460 Compatibility feature. This no-charge option is the same feature used by the integrated DOS 1401/1440/1460 emulator program for the Model 135. The 1401/1440/1460 emulator program is relocatable and thus can operate in one or more MFT partitions. Any number of emulator jobs of the same or different types (1401, 1440, 1460) can execute concurrently with System/370 jobs in the same Model 135, subject to the availability of system resources. Emulator and System/370 jobs can be intermixed in the input stream, since emulator job scheduling, initiation, and resource allocation are handled by OS job management routines. I/O operations are handled by OS data management. Emulator jobs are executed by job priority as is any OS job.

Integrated emulation provides a number of advantages over stand-alone emulation that can increase system throughput. The ability to execute 1400-series jobs under operating system control offers emulation users the benefits of multiprogramming and OS facilities. The advantages of Model 135 integrated emulation are:

- Significantly better resource utilization, since System/370 native mode and 1400-series emulator jobs can be multiprogrammed. Stand-alone emulators normally use only a portion of the hardware resources available in the system.

- Standardized and simplified job accounting and job scheduling. The latter reduces the number of IPL's required because switching from operating system to stand-alone emulation mode of system operation is not required.

- The ability to extend or add applications such as graphics, teleprocessing, time sharing, etc., because a dedicated emulation environment is no longer required and more system resources are available in a given time period.

- The ability to process certain 1400-series tape and disk data sets, using both System/370 and emulated 1400-series programs. Existing 1400 tape files can be converted to a standard OS data format using the IBM-supplied Tape Preprocessor formatting program. Existing 1400 disk files must be converted.

- More efficient use of direct access space, since both 1400 and System/370 data sets can be placed on the same disk volume.

- Device independence for emulator-supported I/O devices used by emulated 1400-series programs that handle data sets in OS VBS data format. OS access methods are used to handle I/O operations so that new functions and I/O device support added to the access method routines used by the emulator programs are automatically available to emulated 1400-series jobs. Tape and unit record 1400-series files can be emulated on System/370 direct access devices.

The emulator program provided for the Model 135 uses simulation routines, compatibility feature microprogram instructions, the Model 135 instruction set, and OS MFT supervisor and data management routines to emulate 1400-series program operations. QSAM is used for unit record emulation, BSAM is used for tape emulation, and BDAM is used for disk emulation. Figure 40.10.1 shows the general layout of an emulator program partition and indicates the range of processor storage requirements for the 1400 emulator program. Note that 1400 storage is simulated in contiguous Model 135 storage (each 1400 position is simulated by a byte).

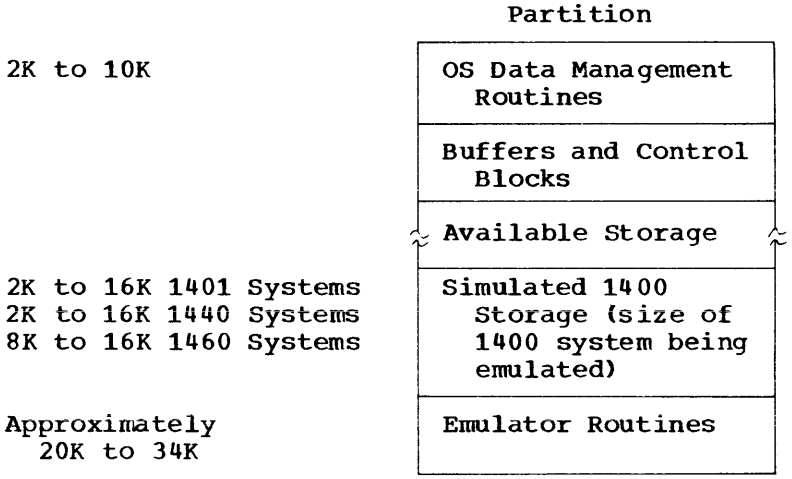


Figure 40.10.1. Partition layout for an OS MFT 1400-series emulator program job step, with general storage requirements indicated

The specific emulator program to be used by an installation must be constructed via an emulator generation procedure, which produces

the job control statements required to assemble and link-edit the desired emulator modules and place the emulator program in SYS1.LINKLIB. Emulator program routines and data formatting programs (Tape Preprocessor, Tape Postprocessor, and Format Disk) are distributed on a restore tape independently from regular OS releases. Model 135 1400-series emulator users receive two tape formatting programs and one disk formatting program. The following must be done to include an emulator in an OS MFT operating system for the Model 135:

- Certain facts about the emulator program to be used with the operating system generated must be specified in the input required to generate the OS control program.

- The emulator restore tape must be obtained from PID and an emulator program with the desired facilities must be generated and placed in SYS1.LINKLIB. More than one version of a 1400-series emulator program for the same and different 1400 systems can exist in an operating system.

The emulator program generated will emulate, without change, 1400 programs that are written in accordance with IBM 1400 Principles of Operation manuals and that are operating on 1400 systems, subject to the following conditions:

1. Time-dependent programs may not execute properly. Provision has been made to allow some time-dependent programs to be emulated correctly. (See the appropriate emulator planning manual for details.)

2. Programs that depend on error conditions or on the absence of a particular feature may not be emulated correctly.

3. Programs with undetected programming errors will give unpredictable results.

4. Only the 64-character BCD set is accepted by the emulators.

5. Programs that use unsupported features or I/O devices (as described in this subsection and in the OS 1400 emulator program reference manual) must be modified to conform to the support provided by the specific emulator program unless a user routine is written to handle the feature.

The Model 135 1400 integrated emulator program supports the same facilities as the stand-alone 1401/1440/1460 emulators for System/360 models except for a few special features not supported by the Model 135 emulators. Thus, any 1400 program that is being executed by a stand-alone System/360 emulator and that does not use one of these special features can be emulated on the Model 135 without change.

Tape and Disk Formatting Programs and Data Formats

The Tape Preprocessor and Tape Postprocessor formatting programs supplied to Model 135 1400-series emulator users operate as processing programs and can be executed with any OS control program generated with the emulator macro specified. The Tape Preprocessor operates in a program area of 4K bytes plus I/O buffer requirements and accepts as input seven- and nine-track tape in 1400-series format with or without 1400 labels. It accepts mixed-density 1400 files, that is, files with header labels written in a density different from that of the data. The emulator program considers a change in density to be an error and expects the emulated program to handle this condition. Therefore, it may be desirable to preprocess mixed-density 1400-format files.

The preprocessor produces as output spanned variable-length (OS VBS) format data that can be written on seven- or nine-track tape or on direct access storage. Input records longer than 32,755 bytes are reblocked, since OS BSAM cannot handle a physical data block longer than 32K. VBS format tape can be unlabeled or have OS standard labels, in addition to any 1400 labels. The preprocessor converts 1400 labels and tapemarks into data records that are recognized by the emulator program. Thus, if VBS format tapes with 1400 label data records are to be processed by System/370 programs, the tape must be rewritten to remove the label data records or the System/370 program must contain a routine to recognize these records.

The Tape Postprocessor operates in a program area of 5K bytes plus I/O buffer requirements and performs the reverse of the Tape Preprocessor. The postprocessor program is useful when a copy of a data set in OS VBS and another in 1400 format are required or if mixed-density 1400 format files are required. (The 1401/1440/1460 emulator program accepts as input and produces as output both the formats handled by these two tape formatting programs.)

The tape formatting and emulator program handle 200-, 556-, 800-, and 1600-BPI-density, mixed-density, and even, odd, and mixed-parity seven-track tapes. VBS or 1400-format data written on nine-track tape is coded in EBCDIC. If VBS format tapes are processed on a seven-track tape unit, the tape control unit must have seven-track and data convert features installed. The alternate mode used by stand-alone System/360 1400 emulators is accepted by the Model 135 1401/1440/1460 emulator program as well.

While existing tape files with blocks longer than 32K bytes must be preprocessed, conversion to VBS format offers the following advantages:

- The ability to emulate tape data sets on direct access devices for more flexibility in I/O device assignment
- The ability to increase emulator job performance by reblocking 1400-series-format tape files with short blocks
- The ability to reduce processor storage buffer requirements by reblocking files with very large blocks
- The ability to process VBS format tape and move mode disk data sets with both OS and emulated 1400-series programs if the OS programs can handle 1400 label and tapemark records. (Load mode disk data sets are not accepted by OS programs.)

The disk formatting program supplied operates as a processing program, and an area of 2K bytes in addition to buffer requirements is needed for its execution. This program must be used to format System/370 disk volumes that are to contain 1400 disk data. In order to convert 1400 disk files that are being processed on a 1400 system or by a stand-alone System/360 1400 emulator to a format acceptable to the Model 135 1401/1440/1460 emulator program, the following must be done:

1. The contents of the disk must be dumped to tape, using a 1400 disk-to-tape utility program. This must be done on a 1400 system if the data was created and is being processed on a 1400 system. A System/360 with a stand-alone 1400 emulator must be used to convert 1400-format data files that are being emulated on System/360 direct access devices.

2. One or more initialized System/370 disk volumes must be formatted by executing the IBM-supplied disk formatting program on a Model 135 under OS MFT control.

3. A 1400 tape-to-disk utility program must be executed on a Model 135 under 1400 emulator program control to restore the 1400 data on tape to the formatted System/370 disk volume(s).

The data records on one 1400 disk track are formatted into one fixed-length record, which is generally placed on one System/370 disk track. If one System/370 track is not large enough to contain the fixed-length record created from one 1400 track, the track overflow feature is required on the System/370 device or another System/370 disk device type must be used. Depending on the disk devices involved, one System/370 disk track may be large enough to contain the data from more than one 1400 disk track, and use of the Track Overflow feature is an option. If used, it may result in more efficient use of System/370 direct access space. However, I/O processing time is increased by use of the Track Overflow feature.

Job Submission and Operator Communication

OS job control and Model 135 1400 emulator program control statements must be present for each emulated 1400-series object program. Subject to the restrictions listed previously, existing 1400 programs need not be modified.

The required emulator control statements for emulated 1400 programs are provided in a card, tape, or disk sequential data set or in the input stream. The 1400 object programs to be emulated can be placed in the input stream, on tape, or in a partitioned data set on disk. An emulator control statement describes their location.

Card input to 1400 programs can be placed in the input stream to be read by the reader/interpreter and placed in a SYSIN data set. Alternately, card input can be emulated via a tape or disk sequential data set. Data to be printed or punched can be placed in a sequential tape or disk data set or in a SYSOUT data set on disk for transcription by an output writer. Use of an OS reader interpreter and output writer to handle unit record operations should reduce the execution time required to emulate 1400 programs.

The operator communicates with an emulator partition via emulator commands that can be entered using the operating system console. The commands provided allow simulation of 1400 console operations and can also be used in debugging operations. The operator can display I/O assignments, sense switch settings, etc., in effect for an executing 1400 program. In addition, the operator can alter and dump processor storage selectively within the emulator program partition.

If multiple console support (MCS) is included in the OS control program generated, emulator program messages can be routed to a specific console device so that emulation messages are isolated.

Installing an Emulator

The following outlines the general procedure required to make the transition from 1400 system operation or stand-alone System/360 1400 emulation to Model 135 emulation under OS MFT.

1. Generate an OS MFT operating system for the Model 135 and specify required parameters that describe the 1400 emulator program to be used with this operating system. (Generating an operating system for the Model 135 is discussed in Section 60.)

2. Generate the desired 1400 emulator program.

3. Add required OS job control and emulator control statements to the existing 1400 programs that are to be emulated on the Model 135. Subject to the conditions stated previously in this section, modification of 1400 programs may or may not be required. Optionally, 1400 programs can be placed in a library (PDS) by using the OS IEBGENER utility program.

4. Tapes in 1400 mode with blocks shorter than 18 bytes or longer than 32K must be preprocessed.

5. Disk files must be transferred to Model 135 direct access devices by using the steps already outlined.

6. Optionally, routines to support features and I/O devices not handled by the emulator program can be written and placed in a library. The generated 1400 emulator program will cause them to be loaded.

When installation of a 1400 emulator is being planned, consideration should be given to the factors that affect the performance of emulated 1400 jobs. Throughput of 1400 jobs is affected by the mix of CPU and I/O operations executed by the compatibility feature and the amount of interference from higher priority partitions. A large factor in performance is the way I/O operations are handled. The following steps can be taken to achieve improved emulator job throughput if enough processor storage is available:

1. Allocate one buffer to each access mechanism simulated instead of sharing buffers among multiple access mechanisms. (Do not use the shared buffer option for disk data sets unless processor storage is limited.)

2. Allocate two buffers to each tape data set instead of one.

3. Convert 1400-format tape files to VBS format for emulator processing.

4. Using the Tape Preprocessor, reblock tape files containing short blocks.

5. Do not use the Track Overflow feature for emulated disk data sets unless direct access space is at a premium.

SUPPORT OF 1401/1440/1460 FEATURES

The OS 1401/1440/1460 Emulator program can operate on a Model 135 with 144K or more of storage, the 1401/1440/1460 Compatibility feature, and enough I/O devices for the operating system and emulated 1400 devices.

The partition size required depends on the features, I/O devices, and buffering used and on the size of the system being emulated. The emulator program size varies from a minimum of approximately 20K, for a basic system with unit record I/O device support only, to a maximum of approximately 34K for all special features and unit record, tape, and 1311/1301 disk support. A 16K 1401 system with unit record devices (400 bytes of buffers) and six tape units (1K buffer per tape unit) can be emulated in a 54K partition. This figure includes approximately 3.9K for access method routines. (For details about processor storage requirements, see Emulating the IBM 1401, 1440, and 1460 on IBM System/370 Models 135, 145, and 155 using OS/360.)

Note that a 1401/1440/1460 Emulator program can be generated to handle 1405 Disk Storage or 1311/1301 Disk Storage but not both.

Table 40.10.1 lists the 1401/1440/1460 system features supported and not supported by the Model 135 OS 1401/1440/1460 Emulator program. Table 40.10.2 lists 1401/1440/1460 I/O devices and features emulated and their Model 135 I/O device counterparts, while Table 40.10.3 indicates unsupported 1400-series devices.

The number of Model 135 direct access devices required to emulate a 1400-series disk device is indicated in Table 40.10.4. Requirements with and without use of the Track Overflow feature are indicated. A pair of numbers is given in each column for a device. The top entry represents the number of drives required. The bottom entry indicates the number of unused cylinders in the last disk pack used, all others being full (199 cylinders per 2311 and 2314, 403 per 3330).

The Model 135 integrated OS 1401/1440/1460 Emulator program supports the same facilities and I/O devices as the stand-alone 1401/1440/1460 emulator for System/360 models except for 51-Column Card, Punch Feed Read, and Selective Stacker on the 1402 Card Read Punch and Column Binary and Binary Transfer in the CPU.

Table 40.10.1. 1401/1440/1460 system features supported by the Model 135 OS 1401/1440/1460 Emulator program

All basic CPU operations	Sense Switches
1401 Models A-F, H	Indexing and Store Address
1440 Model A	Register (1440,1460)
1460 all models	Advanced Programming
Core storage up to 16,000	Bit Test
positions	Print Storage
Expanded Print Edit	Additional Print Control
Inverted Print Edit	Space Suppression
High-Low-Equal Compare	Processing Overlap
Multiply-Divide	

Table 40.10.2. 1401/1440/1460 I/O devices and features emulated by the OS 1401/1440/1460 Emulator program and their Model 135 counterparts

1401/1440/1460	Model 135 I/O Device
• 1402, 1442 Card Read Punch 1442 Card Reader The following are not emulated: Column Binary (1402) 51-Column Card (1402) Punch-Feed Read (1402) Read-Punch Release (1402) Card Image (1402) Punch-Column-Skip (1442) Read and punch same card (1442) Stacker Selection (treated as a no-op) Programmed reading from more than one reader or punching on more than one punch within a program is not supported.	• Any card reader, card read punch, magnetic tape unit, or direct access device supported by OS QSAM
• 1443, 1403 Printer The Selective Tape Listing feature and programmed printing on more than one printer are not supported.	• Any printer, magnetic tape unit, or direct access device supported by OS QSAM. A printer must have the UCS feature to emulate the preferred character set and numerical print features correctly.
• 729 Model II, IV, V, and VI Magnetic Tape Units 7330, 7335 Magnetic Tape Units (Compressed tapes are not supported.)	• Any tape unit or direct access device supported by OS BSAM. VBS format is used for 1400 tape files emulated on a direct access device. Seven-track tapes must be emulated on tape units with a seven-track head attached to a tape control unit with seven-track and data convert capability
• 1407 Console Inquiry Station 1447 Console The Branch on Buffer Busy feature (1447) is not supported.	• Any Model 135 operator console supported by OS MFT
• 1301 Disk Storage (only one access arm) 1311 Disk Storage Drive 1405 Disk Storage Models 1 and 2 Up to five 1311 drives and one 1301 module or one 1405 Model 1 or 2 can be emulated. All features are supported except Write Disk Check (treated as a no-op). Write verification can be requested in the OS job control statement for the emulated disk device.	• Any direct access device supported by OS BDAM. If two or more System/370 direct access devices are required to emulate one 1400 disk device, all Model 135 disk devices used to emulate that device must of the same type.

Table 40.10.3. 1401/1440/1460 I/O devices and features not supported by the Model 135 OS 1401/1440/1460 Emulator program

1400 I/O Device	1400 Feature
1404 Printer 1444 Card Punch 1445 Printer 1011 Paper Tape Reader 1012 Tape Punch Optical readers Magnetic character readers 7340 Hypertape Drive Teleprocessing devices	1401 Model G Binary Transfer

Table 40.10.4. Model 135 direct access device requirements for emulation of 1401/1440/1460 disk devices using OS with and without the Track Overflow (T.O.) feature. Number of packs required and number of remaining available cylinders on the last pack are shown by the first and second rows of figures, respectively, for each entry. Two figures shown in the number of System/370 drives column indicate that more than one 1400-series device can be emulated on a single Model 135 disk drive.

1401/1440/1460 Disk Device	Number of Model 135 Drives Required per 1400 Device					
	2311 Disk Drives		2314 Disk Drives		3330 Disk Drives	
	Without T.O.	With T.O.	Without T.O.	With T.O.	Without T.O.	With T.O.
1405 Disk (Model 1)	2 64	2 82	2:1 31	2:1 35	8:1 19	8:1 22
1405 Disk (Model 2)	4 131	4 160	1 31	1 35	4:1 2	4:1 22
1311 Disk (Sector mode)	1 99	3:1 6	11:1 12	11:1 12	38:1 3	43:1 4
1311 Disk (Track mode or both track and sector mode)	1 99	2:1 24	7:1 24	8:1 22	30:1 8	33:1 0
1301 Disk (Sector mode)	6 194	4 158	1 32	1 37	3:1 87	4:1 36
1301 Disk (Track mode or both track and sector mode)	6 194	4 48	2 148	1 10	3:1 7	3:1 77

40:15 OS DOS EMULATOR PROGRAM

The availability of the OS DOS emulator offers current DOS users who upgrade to a Model 135 the opportunity to convert to an OS MFT operating environment more easily than is possible without the use of emulation. In addition, the OS DOS emulator user can benefit from

the use of integrated emulation, since the OS DOS emulator can execute concurrently with other OS jobs.

The OS DOS emulator for the Model 135, which is the same DOS emulator provided for System/370 Models 145 and 155, is a combination of the OS DOS emulator processing program and the standard OS/DOS Compatibility feature. This feature provides the relocation necessary for execution of a DOS supervisor and DOS programs under OS control in any processing program storage location. An OS MFT control program generated for a Model 135 is required also.

The OS DOS emulator and the DOS system being emulated (DOS supervisor and up to three processing program partition,s) execute together in an MFT partition which must be a minimum of 38K. The OS DOS emulator program and tables require 22K plus another 4K if I/O staging is used. Additional OS DOS emulator program storage may be required depending on the I/O devices used. Up to ten I/O devices are supported in 22K, and 250 bytes are required for each additional device. The I/O staging requirement of 4K supports unblocked reader, printer, and punch records and residence of the required QSAM routines in the OS DOS emulator partition.

The DOS system being emulated can be 16K, 24K, or 32K and up, in 4K increments. The OS DOS emulator is scheduled to operate in the same manner as any other OS job, and one or more OS DOS emulator jobs can execute concurrently with OS jobs if enough I/O devices and processor storage are available. In addition, the Model 135 OS 1401/1440/1460 Emulator program can execute concurrently with the OS DOS emulator if enough resources are present.

The user need not make any changes to the existing DOS supervisor, job control statements, tape files, or disk files in order to use the OS DOS emulator. Modification of existing DOS programs is required only for programs that contain features unsupported by the OS DOS emulator.

The major advantages of the OS DOS emulator are:

- Transition from a DOS to an OS operating environment is smoother. The conversion of DOS source programs, job control, and data files to OS formats can be done gradually for emulated DOS jobs.

- Model 135 OS DOS emulator users can continue to use most IBM-supplied application programs (Type II and program products) that operate under DOS but not OS and do not use BTAM or QTAM, by emulating them under OS.

- Dedicated emulation is not required, thus allowing the user to take advantage of OS facilities.

- Total system throughput can be increased by operation of the OS DOS emulator in a multiprogramming OS environment and by using the staged I/O option of the OS DOS emulator. The latter permits emulated DOS programs to use the data transcription facilities of the OS reader interpreter and output writer to handle their unit record functions. Use of the staged I/O option of the OS DOS emulator also eliminates the necessity of dedicating unit record devices to DOS emulation.

All operating environments, control program facilities, and I/O devices supported by DOS can be emulated, with the following exceptions:

- Autotest

- OLTEP (which does not produce meaningful results)

- Model-dependent functions such as CS/30 and the DIAGNOSE instruction. (1400 emulation can be handled using the Model 135 OS 1401/1440/1460 Emulator program.)
- Emulation of emulators that operate under DOS, for example, Model 135 integrated 1400 emulators that operate under DOS
- 1259, 1412, and 1419 Magnetic Character Readers
- 1287 and 1288 Optical Character Readers in document mode, if response times are required for pocket selection
- Teleprocessing devices (including the 2260 Display Station)
- Storage protection within the DOS system being emulated (among the DOS supervisor and partitions)
- DOS (and OS) direct access volumes having nonunique volume serial numbers online concurrently

In addition, executable DOS programs cannot be handled that:

- Rely on known timing relationships of the DOS system
- Depend on HALT I/O, READ DIRECT, WRITE DIRECT, and DIAGNOSE instructions for their operation
- Require more than two bytes of sense data for an I/O device
- Depend on the PCI flag in a CCW (the flag is ignored by the emulator)
- Modify or use information in CCW's after the CCW list is initiated with a START I/O instruction and before the I/O operation terminates
- Initiate the same CCW list for an I/O operation on more than one I/O device concurrently

While a pseudo interval timer is maintained at simulated DOS decimal location 80, accurate time of day is not guaranteed, because the timer is running only when the OS DOS emulator partition is executing, and a time lag occurs during the interval required to update the timer.

The following I/O devices are supported by the OS DOS emulator:

- 1052 Printer-Keyboard (emulated on a 3210 Model 1 or 3215 console)
- 1285, 1287, 1288 Optical Readers (the latter two not in document mode)
- 1403, 1443, 1445, 3211 Printers
- 1404 Printer (emulated on a 1403 or 3211 Printer for continuous form operations only)
- 1442, 2520, 2540 Card Read Punches and 2501 Card Reader
- 2311 Disk Storage Drive
- 2314 and 2319 Disk Storage
- 2321 Data Cell Drive
- 2400- and 3400-series magnetic tape units
- 2671 Paper Tape Reader
- 3330 Disk Storage
- 3505 Card Reader
- 3525 Card Punch
- Any new devices that are supported by both DOS and OS, subject to the programming restrictions stated above

EMULATOR JOB SUBMISSION AND GENERAL OPERATION

DOS emulation is initiated as a single-step OS job via the input stream. An OS DOS emulator job can consist of one or more DOS jobs. The OS DOS emulator program, which must reside in SYS1.LINKLIB or a user job library, is specified in the EXEC job control statement included in the job control for the OS DOS emulator job. The following also must be identified in the DD job control statements for the OS DOS emulator job:

1. The DOS system residence and operator console devices
2. The location(s) of the DOS input stream(s)
3. I/O assignments for the staging of DOS unit record I/O operations
4. All the I/O devices that will be used by the DOS programs that are emulated as part of this DOS emulator job

The same device types currently being used by DOS programs must be used when these programs are emulated on the Model 135, except for staged unit record I/O devices. However, the devices used in emulation need not have the same device addresses on the Model 135 as on the current system.

The DOS system background partition input stream can be located in the OS input stream or on a separate data set. DOS batch-initiated foreground (BJF) partition input streams must be located in separate data sets.

If enough Model 135 processor storage is available, I/O staging can be used to increase OS DOS emulator job throughput and reduce the number of devices that have to be dedicated to the DOS emulation partition. It allows DOS unit record files SYSRDR, SYSPCH, and SYSLST to be emulated on direct access devices, using the OS reader interpreter and output writer. DOS job control statements (for the BG partition) and/or card input to DOS programs to be emulated can be placed in the OS input stream and will be transcribed by the reader interpreter to SYSIN data sets on direct access devices. Thus, emulated DOS job steps will obtain their card input from OS SYSIN disk data sets. Output from emulated DOS programs can be placed in OS SYSOUT data sets on disk to be transcribed to the printer or punch by an output writer.

The following should be noted about the use of I/O staging. In OS, a job is not placed in the input queue, from which all jobs are scheduled, until the entire job (job control and input stream data for the job) has been read by the reader interpreter. Similarly, SYSOUT data sets produced during job step execution are not placed in the output queue for transcription by an output writer until job termination.

Thus, if all DOS jobs to be emulated are grouped together as a single OS DOS emulator job, DOS emulation cannot begin until all DOS jobs (and their input stream data) have been read by the reader interpreter, and none of the SYSOUT data sets from completed emulated DOS jobs can be transcribed until the OS DOS emulator job itself terminates (all DOS jobs processed). This negates one advantage of I/O staging, which is the overlapping of unit record input and output data transcription with processing.

Therefore, consideration should be given to grouping DOS jobs into two or more OS DOS emulator jobs that execute one after the other in the OS DOS emulator partition. In addition, if the output from a particular DOS job is desired immediately, it should not be staged (written to a SYSOUT data set). The use of multiple OS DOS emulator jobs, instead of one, in an OS DOS emulator partition offers an additional advantage in optimizing device usage, as discussed below.

I/O operations and I/O error recovery procedures for emulated DOS programs are handled by the OS control program. All I/O devices to be used by emulated DOS programs must be allocated to the DOS emulation partition when the OS DOS emulator job is begun. These devices are dedicated to DOS emulation and cannot be allocated to any other executing OS jobs while DOS emulation is in operation. Thus, direct access devices and their data sets cannot be shared by an OS job step and an emulated DOS program. However, DOS direct access volumes can be shared by DOS partitions being emulated in the same OS DOS emulator partition. In addition, the user must ensure that all online OS DOS emulator direct access volumes have unique volume serial numbers with respect to other DOS and OS direct access volumes online at the same time.

Consideration should be given to grouping DOS jobs into multiple OS DOS emulator jobs according to the types and total number of I/O devices required. This can reduce the number of I/O devices that have to be dedicated to a DOS emulation partition at any given time, thereby making more devices available to other OS jobs.

When the DOS emulation job is initiated, the DOS emulator program is loaded into the OS DOS emulator partition. The DOS emulator program performs control block and table initialization and initiates an IPL from the DOS system residence volume. Once the DOS supervisor has been loaded and has established the DOS partitions, DOS job execution begins. DOS programs are loaded into the defined DOS partitions and emulated. Messages to the operator from the DOS emulator program are issued in standard OS format and include a unique identification to indicate that they are OS DOS emulator messages. If the MCS option is included in the OS control program, all DOS emulation messages can be routed to a specific console, and thus isolated.

The entire OS DOS emulator partition operates with a nonzero storage protect key to prevent it from interfering with the OS control program and other executing OS jobs. Therefore, the DOS emulator program, the DOS supervisor, and other DOS jobs in the emulator partition are not protected from inadvertent modification by an executing DOS program.

INSTALLATION OF THE OS DOS EMULATOR

The following are the major steps that a DOS user must take to install the OS DOS emulator on a Model 135:

- Data processing personnel--systems analysts and designers, programmers, operators, etc.--must be educated on OS.
- The installation must decide which functions and options are to be included in the generated OS MFT control program.
- The desired OS MFT operating system must be generated using a release of OS that includes Model 135 support. The DOS emulator option must be requested. Installation-designed routines, such as nonstandard tape label processing, accounting, etc., must be written, as required, and added to the generated operating system.
- The DOS emulator must be obtained (it is distributed separately from OS) and added to the generated operating system.
- DOS jobs that cannot be emulated must be converted to OS format. This involves source program changes, conversion of DOS job control statements to OS job control statements, and, depending on data organization, conversion of DOS files to OS data sets. The amount of reprogramming required depends on the source language being

used. In general, high-level language programs require much less modification than Assembler Language programs.

- The volume serial numbers of all existing DOS direct access files must be inspected for duplicates, and unique serials should be assigned where necessary. Volume serial numbers assigned to newly created DOS files or OS data sets should be unique as well.

- The OS job stream should be planned, and consideration should be given as to how OS DOS emulator jobs are to be scheduled and executed, as discussed previously in this subsection. Note also that the total storage size of the DOS system being emulated may be reduced. For example, if one DOS processing partition is devoted to teleprocessing or CS/30, which are not emulated by the OS DOS emulator, this DOS partition is no longer required and its storage can be made available for allocation to an OS partition.

- Optionally, the size of the emulated DOS system can also be reduced by the removal of functions that are now provided by OS. For example, DOS POWER can be removed from a DOS system, since data transcription can be handled by the OS reader interpreter and output writer. The model-dependent DOS MCRR routine can be removed from a DOS supervisor, as Model 135 MCH and CCH routines contained in the OS control program will be used for machine and channel error handling.

  Note that alterations affecting the DOS supervisor normally require a system generation to be performed. In addition, any change resulting in a different starting address for a DOS partition means that existing nonrelocatable DOS programs executing in that DOS partition must be link-edited relative to the new address.

Figure 40.15.1 illustrates a 192K Model 135 configuration that supports one OS processing partition (P3), a transient or resident 32K reader interpreter (in P2), a resident output writer (P0), and emulation of a 64K DOS system (P1), of which only 56K need be emulated. The staged I/O option is not used.

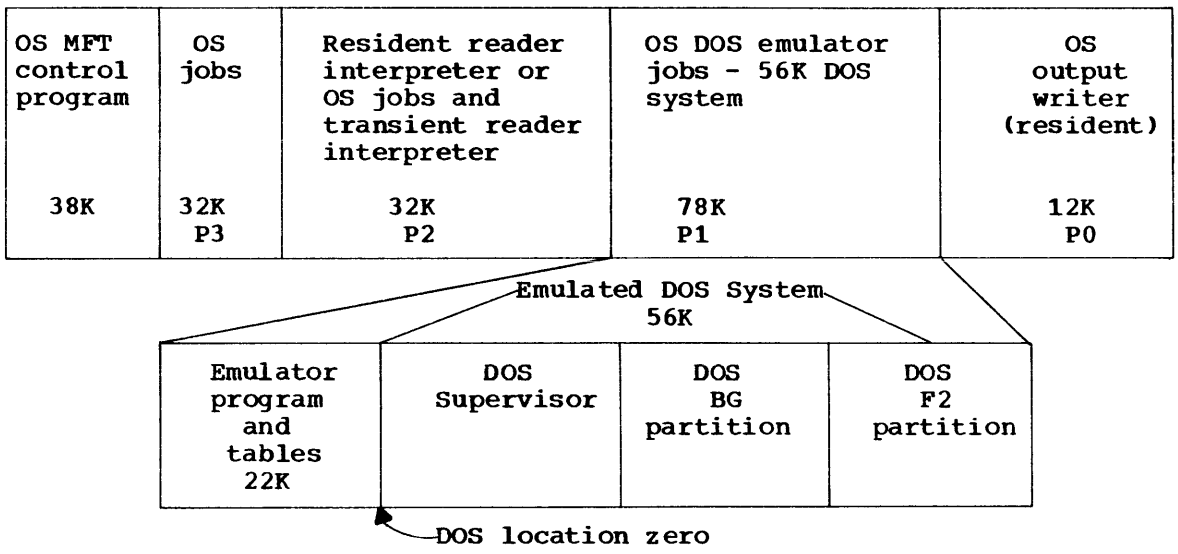


Figure 40.15.1. Sample 192K Model 135 configuration for emulation of a 56K DOS system

SECTION 50: RELIABILITY, AVAILABILITY, AND SERVICEABILITY (RAS) FEATURES

50:05 INTRODUCTION

With the growth of more and more online data processing activities, as distinguished from traditional batch accounting functions, the availability of a data processing system becomes a very essential factor in company operations, and complete system failure is extremely disruptive. Because of the growing frequency of online processing and the fact that the Model 135 is designed to operate in such an environment, IBM has provided an extensive group of advanced reliability, availability, and serviceability features for the System/370 Model 135. These RAS features are designed to improve the reliability of system hardware, to increase the availability of the computing system, and to improve the serviceability of system hardware components.

The objective of the RAS features of the System/370 Model 135 is to reduce the frequency and impact of system interruptions that are caused by hardware failure and necessitate a re-IPL. RAS features are as follows:

- Hardware reliability is enhanced through use of more reliable components.
- Recovery facilities, both hardware and programming systems, not available for System/360 Models 25 and 30, are provided to reduce the number of failures that cause a complete system termination. This permits deferred maintenance.
- Repair procedures include use of predictive maintenance routines to identify system malfunctions that can be repaired during a scheduled or deferred maintenance period. In addition, when necessary, online diagnosis and repair of malfunctions concurrent with normal job execution in a multiprogramming environment can be performed in order to reduce the effect of such repairs on system unavailable time.

Each RAS feature, recovery or repair, is discussed in the remainder of this section.

The following recovery features are implemented in hardware:

- Automatic retry of most instructions when a CPU error occurs during their execution
- ECC validity checking on processor and control storage to correct all single-bit and detect all double-bit errors
- I/O operation retry facilities, including channel retry data provided in the limited channel logout area, called the extended channel status word (ECSW), and channel/control unit command retry procedures to correct failing I/O operations
- Expanded machine check interrupt facilities to facilitate better error recording and recovery procedures

The following recovery features are provided by programming systems:

- Recovery management support (RMS) to handle the expanded machine check interrupt and channel retry data. MCAR and CCH routines are provided for DOS. MCH and CCH routines are provided for OS (MFT only for the Model 135).
- Error recovery procedures (ERP) to retry failing I/O device and channel operations (DOS and OS)
- OBR and SDR routines (DOS and OS) to record statistics for I/O errors
- Environment recording, edit, and print program (EREP) for DOS and OS to format and print error log records
- I/O RMS routines (OS)--alternate path retry (APR) and dynamic device reconfiguration (DDR)--to provide additional recovery procedures after channel or I/O device failures
- Checkpoint/restart (DOS and OS) and warm start facilities (OS) to simplify and speed up system restart procedures after a failure necessitates a re-IPL

The following repair features are provided:

- Online Test Executive program (OLTEP) and Online Tests (OLT's) that execute under operating system control (DOS and OS) and provide online diagnosis of I/O device errors for most devices that attach to the Model 135
- Diagnostics to locate the malfunctioning field-replaceable unit

These aids are designed to enhance system availability. In many cases, the system can run in a degraded mode so that maintenance can be deferred to scheduled maintenance periods. When solid failures do occur, their impact can be reduced by faster isolation and repair of the malfunction than is currently possible on System/360.

50:10 RECOVERY FEATURES

Additional hardware, which attempts correction of most hardware errors without programming assistance, has been included as part of the basic Model 135 system. The control program can be notified, via an interrupt, of both intermittent and solid hardware errors so that error recording and recovery procedures can take place.

AUTOMATIC INSTRUCTION RETRY

If a CPU hardware error is detected during the execution of an instruction, it is retried automatically by the system, without programming assistance, if certain conditions exist. Specifically, retry occurs if the instruction that failed is a LOAD CONTROL or LOAD PSW that did not pass beyond an established threshold point during execution or, for other instructions, if the error occurred before any processor storage location, general register, or floating-point register was changed. If alteration has occurred, the retry microprogram routine, which is part of the basic system microcode, performs an analysis to determine whether source operands have been changed. If they have not, the instruction is retried from the beginning up to eight times before it is determined that the error is uncorrectable. The percentage of retry effectiveness is reduced if a program frequently uses instructions that change the source operand

early during instruction execution (specifically, TRANSLATE, TRANSLATE AND TEST, EDIT, EDIT AND MARK, and EXCLUSIVE OR CHARACTERS).

If an instruction retry is successful, a machine check interrupt is taken, if enabled, so that error recording can be done. If the instruction cannot be retried at all, or if retry is unsuccessful in correcting the error, a machine check interrupt occurs (if enabled) and programmed error recovery procedures should be executed.

The instruction retry feature provides the Model 135 with the ability to recover from most intermittent CPU failures that would otherwise cause a system halt and necessitate a re-IPL or that would cause an executing program to be terminated. Corrected errors are logged for later diagnosis during scheduled maintenance periods, thereby increasing system availability.

ECC VALIDITY CHECKING ON PROCESSOR AND CONTROL STORAGE

The ECC method of validity checking on both processor and control storage provides automatic single-bit error detection and correction. It also detects all double-bit and some multiple-bit processor and control storage errors but does not correct them. Checking is handled on an eight-byte basis, using an eight-bit modified Hamming code, rather than on a single-byte basis, using a single parity bit. However, parity checking is still used to verify other data in a Model 135 system that is not contained in processor or control storage. Models 25 and 30 use parity checking for main storage data verification.

As data enters and leaves storage, ECC logic performs validity checking on each doubleword. When a doubleword (72 bits, as shown in Figure 50.10.1) is fetched from processor or control storage, the 8-bit ECC code is checked to validate the 64 data bits. If the data is correct, the appropriate parity bit for each of the 8 data bytes is generated and the doubleword is reformatted to look as shown in Figure 50.10.2. If a single-bit error is detected, the identified data bit in error is corrected automatically by ECC logic. The corrected doubleword is sent to the CPU but not back to processor storage or control storage. When a doubleword is to be placed in processor storage by a program or in control storage during microprogram loading, the 8 parity bits are removed and the 8-bit ECC code is generated and appended to the 64 data bits. The 72 bits are then stored as shown in Figure 50.10.1.

If a double- or multiple-bit error occurs during instruction execution, the instruction is retried if possible, as already discussed. If the storage error is of an intermittent type and a single-bit error results after any one of the retries, the error is corrected as usual and processing continues.

A machine check interrupt occurs, if enabled, after the occurrence of a double- or multiple-bit storage error and the failing storage address is indicated. However, a single-bit correction in either processor or control storage does not cause a machine check interrupt condition unless the error frequency limit (EFL) of 256 single-bit storage corrections in 416 microseconds is exceeded (and this interrupt is enabled). The hardware maintains an error frequency limit counter, which is incremented each time a single-bit processor or control storage error is corrected and reset to zero every 416 microseconds. When the EFL is reached, a machine check interrupt occurs, when enabled, to inform the operator. The system will continue to correct single-bit errors after the EFL is reached and will cause another interrupt if the EFL is again reached when this interrupt is enabled.

Thus storage repair can be deferred and system operation can continue if necessary.

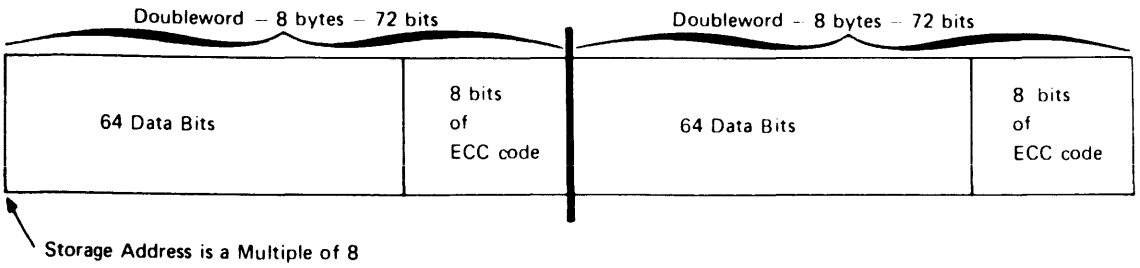
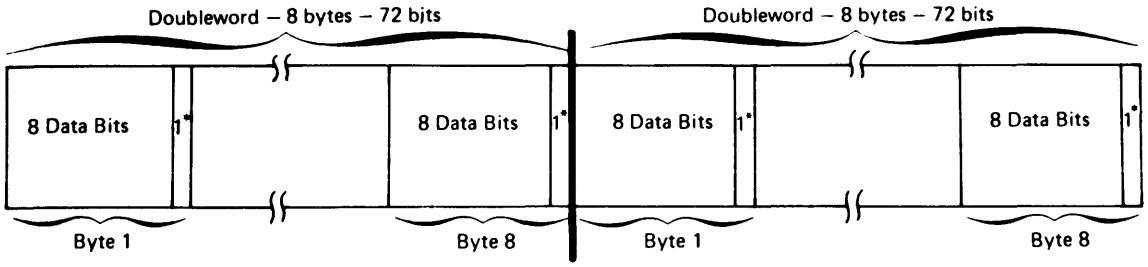


Figure 50.10.1. Data representation used in Model 135 processor and control storage



*Parity Bit

Figure 50.10.2. Data representation used in Models 25 and 30 processor storage and in the Model 135 in other than processor and control storage

The occurrence of an interrupt after the EFL is reached is controlled by the EFL mask bit and the recovery mask bit. (The latter is described under "Expanded Machine Check Interrupt Facilities".) If the recovery mask is disabled, an EFL interrupt does not occur, regardless of the setting of the EFL mask bit. When the recovery mask is enabled, the state of the EFL mask bit, which is set via the DIAGNOSE instruction, determines whether or not an EFL interrupt occurs.

If a machine check interrupt is taken after a double- or multiple-bit storage error, identification of the failing processor or control storage address is provided in a fixed storage area (discussed in the machine check interrupt explanation).

When a double- or multiple-bit processor storage error occurs during an I/O operation, it is reported during the ensuing I/O interrupt so that error recording and I/O retry procedures can be executed. A machine check interrupt is not taken.

The ECC feature increases Model 135 system availability by permitting system operation to continue normally after single-bit processor and control storage errors occur and are corrected. Any processor storage errors on Models 25 and 30 necessitate at least termination of the

processing program involved, since neither hardware nor programmed retry of processor storage errors is provided for these systems. Nor is correction of control storage errors provided.

I/O OPERATION RETRY

Channel retry and command retry features are provided to reduce the number of abnormal program terminations and unscheduled system halts that occur because of I/O errors.

- Channel retry

  This feature has been implemented to ensure that most failing channel operations can be retried by error-handling routines. Both a limited and an extended channel logout are implemented. When a channel error or a CPU error associated with a channel operation occurs, the channel status word (CSW) and a new extended channel status word (ECSW) are stored in the fixed lower storage area (I/O communications area) during the I/O interrupt. The ECSW, or limited channel logout data, provides additional, more exacting status information about the channel failure. The CCH routine passes this data to a device-dependent error recovery routine to be used in the retry of the failing I/O operation.

  Model 135 channels also attempt to log out any time a CSW is stored that indicates an interface control check or a channel control check error. Both CPU and extended channel logout data are stored in the logout area in locations 256-277. The extended channel logout is stored in locations 268-277. This data is to be logged for later diagnosis by customer engineers.

  Channel error retry routines (channel check handlers) for System/360 models are provided only for Models 65 and higher. However, after a channel error occurs, these systems do not always present enough information to the error recovery routines to enable them to retry the failing operation. In other cases, the channel may be left in a condition in which retry is impossible after a channel malfunction. Model 135 hardware improvements eliminate these two situations in most instances.

- Command retry

  Command retry is a channel/control unit procedure that can cause an improperly executed command in a channel program to be retried automatically by hardware so that an I/O interrupt and programmed error recovery are not required. An indication is presented when the control unit recognizes this situation. The byte multiplexer channel will not perform a command retry.

  The command retry feature is implemented in the control unit of the 3330 facility and is discussed in Section 20.

EXPANDED MACHINE CHECK INTERRUPT FACILITIES

Implementation of the machine check level of interrupt on the System/370 Model 135 has been expanded in order to enhance error recording and error recovery procedures. Programming support of the extended machine check interrupt is provided by the Model 135 MCAR and MCH routines of DOS and OS MFT, respectively.

The machine check interrupt facilities of the Model 135 differ from those of Models 25 and 30 as follows:

- Five types of machine check are defined.
- A machine check interrupt occurs to permit the recording of errors corrected by the hardware as well as to allow recovery routines to handle errors that cannot be corrected by hardware.
- Machine check interrupt masking is expanded to handle selective disabling and enabling of the interrupt types defined.
- The size of the fixed area in lower processor storage is increased to accommodate the storing of additional machine status and diagnostic information when a machine check interrupt occurs.
- Hard stop error conditions are defined that cause the Model 135 to stop functioning immediately because the nature of the machine malfunction prevents valid processing from continuing.

The Model 135 presents one of five types of machine check interrupt, depending on the specific machine malfunction, and each type of interrupt is maskable. A machine error causes a soft machine check or a hard machine check interrupt when its type is enabled. A soft machine check occurs after the hardware has been successful in correcting an error or after an error has occurred that does not adversely affect the executing program. This is done so that the failure can be recorded. System operation continues after the error is logged. For example, if an error occurs during the execution of an instruction and if the error is corrected by reexecuting the failing instruction, a soft machine check interrupt occurs at the completion of the successful execution.

A hard machine check occurs when hardware retry fails or is not possible. For example, if an instruction is not executed correctly after eight retries, a hard, rather than soft, machine check interrupt occurs after the last unsuccessful retry.

Figure 50.10.3 shows the layout of fixed processor storage in the Model 135. Fixed storage consists of three areas: the fixed locations in decimal addresses 0-127, the I/O communications area in locations 160-191 (stored during certain I/O interrupts), and the fixed logout area in locations 232-511.

Fixed locations 0-127 are identical in layout and content to these locations in System/360 models, with the exception of the EBCDIC/ASCII bit in the PSW, which must be set to zero.

A logout to the fixed logout area (232-511) occurs when any type of machine check interrupt is taken. The data stored is processed by recovery management routines. The layout of this area is model independent among System/370 models; however, all models do not use every field or bit defined. The fixed logout area data indicates the reason for the interrupt in the machine check code (locations 232-239). The save areas in the logout area preserve the status of the system at the time of the machine check interrupt and contain the contents of the general, the floating-point, and the control registers.

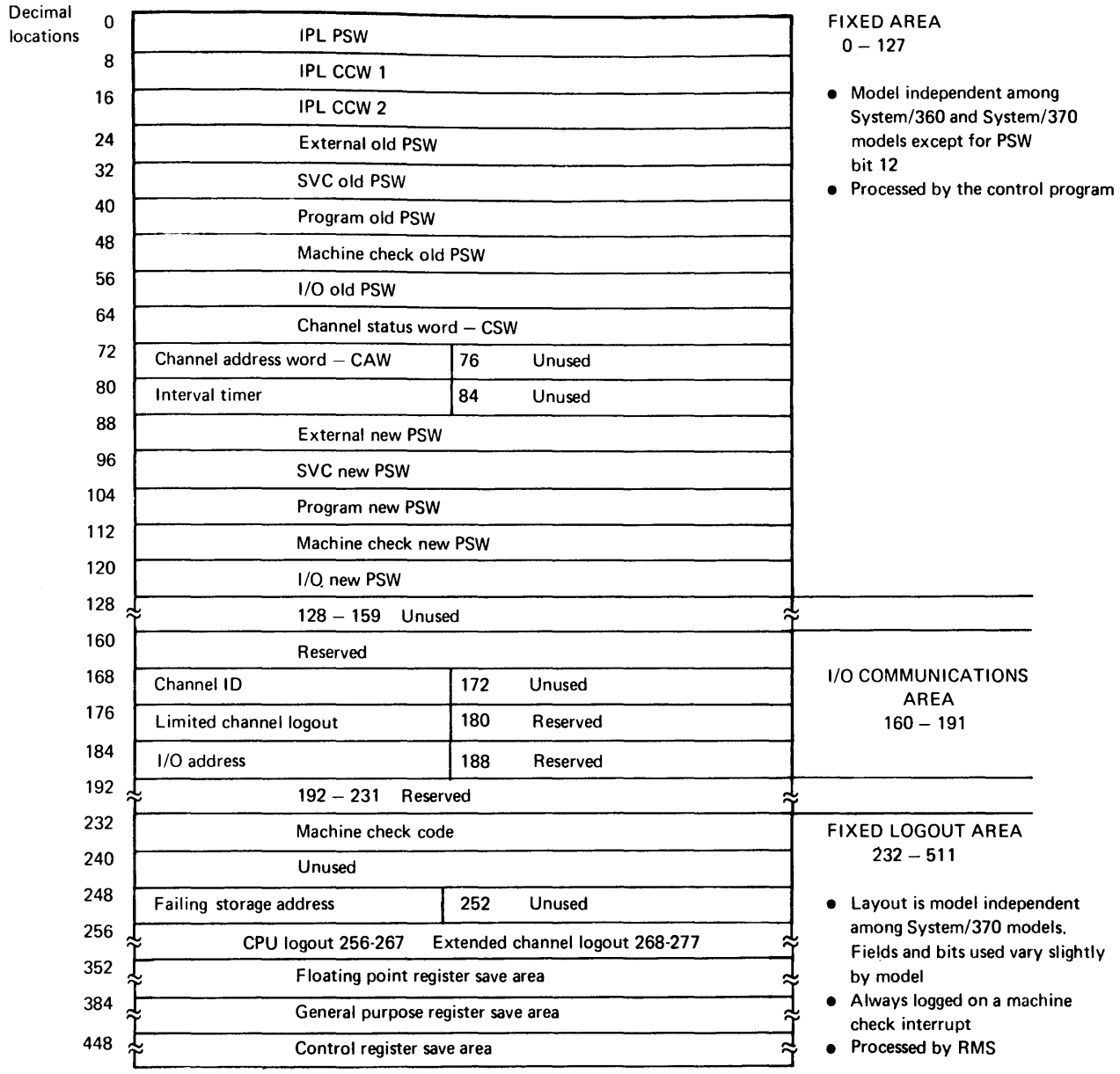


Figure 50.10.3. Model 135 fixed storage locations

Figure 50.10.4 illustrates the layout and the contents of the eight-byte machine check code stored in processor storage locations 232-239. The machine check code indicates which type of interrupt occurred and the validity of certain fields stored in the fixed logout area. Locations 248-251 identify the failing storage address when a double- or a multiple-bit storage error occurs. A bit in the CPU logout data (256-267) indicates whether the error was in control or processor storage.

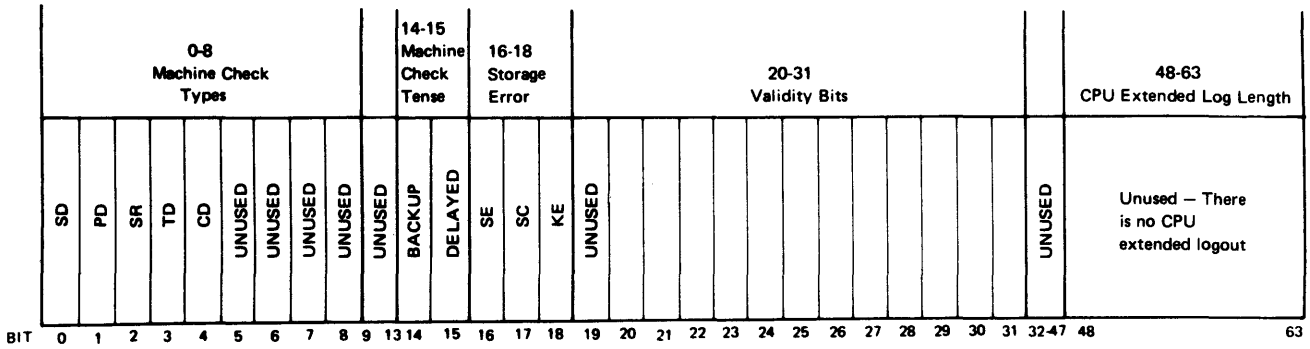
Table 50.10.1 lists the machine check types defined for the Model 135. They are described in the discussion that follows. The mask bits used to enable or disable interrupts for each type are indicated and the setting of the machine check code is discussed. PSW bit 13 and two other mask bits are used to enable and disable machine check interrupts. The recovery mask (R) and external mask (E) bits are

contained in control register 14 and operate subject to PSW bit 13. If PSW bit 13 is disabled, all machine checks are masked. If PSW bit 13 is enabled, the settings of the two additional mask bits determine whether or not interrupts, other than Instruction Processing Damage and System Damage, will be taken. Refer to Figure 50.10.4. (On the Model 135, when a machine check interrupt occurs for an enabled machine check type, all pending machine checks for disabled machine check types are presented also.)

Table 50.10.1. Model 135 machine check interrupts

Mask Bit(s)	Interrupt Type and Cause	Machine Check
PSW 13 and R	System Recovery •CPU error during instruction execution corrected by instruction retry •EFL for single-bit processor and control storage error corrections reached (EFL mask must be enabled also).	Soft
PSW 13 and E	Interval Timer Damage	Soft
PSW 13 and E	Time of Day Clock Damage	Soft
PSW 13	System Damage •Irreparable hardware malfunction	Hard
PSW 13	Instruction Processing Damage One of the following occurs during instruction execution: •Unretryable CPU error •Uncorrectable CPU error •Double- or multiple-bit processor or control storage error •Storage protect key failure	Hard

Fixed Logout Area Locations 232-239



Bit	Interrupt Type
0	SD – System Damage
1	PD – Instruction Processing Damage
2	SR – System Recovery
3	TD – Timer Damage
4	CD – Time of Day Clock Damage

Bit	Storage Error
16	Storage Error Uncorrected
17	Error Frequency Limit
18	Key in Storage Error Uncorrected

Bit	Valid Fixed Area Data
20-23	Machine Check Old PSW (48-55)
	20 AMWP
	21 Masks and Protect Key
	22 Program Mask and Condition Code
	23 Instruction Address
24	Failing Storage Address (248,249)
25	Unused
26	Unused
27	Floating Point Registers (352-383)
28	General Purpose Registers (384-447)
29	Control Registers (448-511)
30	Unused
31	Storage (Validity of processor storage being processed by instructions when interrupt occurred)

Figure 50.10.4. Model 135 machine check code

Soft Machine Check Interrupts

Soft machine check interrupts are as follows:

- **System Recovery. This interrupt occurs if both PSW bit 13 and the recovery mask bit are on. It is caused by a successful instruction retry or having reached the EFL for single-bit processor and control storage error corrections. The EFL mask must also be enabled for an EFL interrupt to occur.**

  The SR bit in the stored machine check code (bit 2) is used to indicate successful hardware recovery. When the SR bit is on without another recovery bit, an error has occurred in the normal functioning of the CPU during instruction execution and the instruction has been retried successfully by instruction retry hardware. CPU logout data is generated and error logging is required.

  The SC bit in the stored machine check code (bit 17) is used together with the SR bit to indicate that the error frequency limit for single-bit processor and control storage errors has been reached. Storage should be repaired during the next maintenance period.

- Interval Timer Damage. This interrupt occurs if PSW bit 13 and the external mask bit are on. It indicates damage to the

timer. Programmed validation procedures and error logging are required.

- Time of Day Clock Damage. This interrupt occurs if both PSW bit 13 and the external mask bit are on. The CD bit in the stored machine check code (bit 4) is used to indicate that an error occurred in the time of day clock that renders the clock invalid. Once this invalid indication has been given, subsequent STORE CLOCK instructions cause the condition code in the current PSW to indicate the fact that the clock is invalid. Error logging is required as a result of clock failure.

Hard Machine Check Interrupts

Hard machine check interrupts are as follows:

- Instruction Processing Damage. This interrupt occurs if PSW bit 13 is on. The PD bit in the stored machine check interrupt code (bit 1) is used to indicate that an error occurred during the execution of the instruction indicated by the machine check old PSW. The error was either a double- or multiple-bit processor or control storage failure, a storage protect key failure, or a CPU error that was unretryable or that could not be corrected by instruction retry processing.

  If a double- or multiple-bit processor or control storage failure caused the interrupt, the address of the failing storage area is indicated in locations 248-251. A bit in the CPU logout area (256-267) indicates whether it was a processor or a control storage error. The KE bit in the machine check code (bit 18) is set on when a storage protection failure occurs. The processor storage block affected is indicated in the failing storage address field (248-251).

  If an unsuccessfully retried CPU failure caused the interrupt, the backup bit in the machine check code (bit 14) indicates the extent of the damage that occurred, if any. If the back-up bit is on, it indicates that no source data has been changed and that the PSW, the registers, and storage reflect the valid state that existed at the beginning of the instruction.

  Error logging and the execution of recovery procedures are required after this interrupt type.

- System Damage. This interrupt occurs if PSW bit 13 is on. The SD bit in the stored machine check code (bit 0) is used to indicate that an irreparable CPU failure occurred that was not a result of the execution of the instruction indicated in the machine check old PSW. An unsuccessful interrupt attempt, control register damage, etc., are examples of system damage errors. System damage also is indicated if the error cannot be identified as one of the other types of machine check interrupt. Programmed error recovery is not possible after this type of failure.

Modes of System Operation for Machine Check Interrupts

Two modes of system operation for machine check interrupts are possible: full recording mode and quiet, or nonrecording, mode. In full recording mode all machine check interrupt types are enabled and cause an interrupt to be taken and logouts to occur. This is the normal mode of Model 135 operation. In quiet mode, all or certain soft machine check interrupts are disabled. Quiet mode can be used to permit system operation without error recording for certain soft machine check errors when the Model 135 is operating under the control

of an operating system without Model 135 machine check handling routines included.

A hard stop status and a hard stop bit have been defined for the Model 135. The hard stop bit is located in control register 14 with the other two mask bits discussed. If a hard stop condition occurs, the Model 135 system ceases all operations immediately without the occurrence of a logout to the fixed area. Hard stop is initiated by hardware rather than by programming.

Generally speaking, a hard stop situation is caused by the occurrence of a hard machine check type of error during the processing of a previous hard machine check error. Implementation of a hard stop prevents system operations from continuing when the nature of the machine malfunction prevents the system from presenting meaningful status data.

The state of the Model 135 after IPL or a system reset is:

1. Recovery reports are disabled. Successful instruction retries and reaching the EFL for single-bit corrections for both control and processor storage do not cause machine check interrupts. (Both the System Recovery and the EFL masks are disabled.)

2. External interrupts are enabled. Interval Timer Damage or Time of Day Clock Damage causes a machine check interrupt.

3. PSW bit 13 normally is set to one by the IPL PSW (it is disabled by system reset) to enable System Damage and Instruction Processing Damage interrupts. Therefore, an irreparable system error, an unretryable CPU failure, an unsuccessfully retried instruction, or a double- or multiple-bit processor or control storage error associated with instruction processing causes a hard machine check interrupt.

4. Hard stop is enabled.

These settings cause the Model 135 to run in quiet mode for hardware-corrected machine errors. If the Model 135 is to operate in full recording mode for machine errors, the appropriate mask bits must be altered by the control program.

MACHINE CHECKS ON SYSTEM/360 MODELS 25 AND 30

A machine check situation in Models 25 and 30 results from hardware detection of a machine malfunction or of a parity error. Bad parity can occur in main storage, in local storage, in a register, in an adder, etc. Error correction is not attempted by Model 25 or 30 hardware when a machine check occurs (except for some instruction retry capability in the Model 30). If the machine check mask in the current PSW (bit 13) is enabled, a machine check causes an interrupt and a diagnostic scan-out occurs, starting at location 128. The number of bytes logged is model dependent.

If the DOS Machine Check Recording and Recovery (MCRR) routine for the Model 30 is present, it gains CPU control after a machine check interrupt, and the error is logged. A retry of the failing operation is not provided by this routine and the affected program is terminated abnormally. If a recovery routine is not present, the system is placed in a wait state when a machine check interrupt occurs. (OS does not provide an SER routine for the Model 30 and DOS does not provide an MCRR routine for the Model 25.)

RECOVERY MANAGEMENT SUPPORT (RMS) FOR DOS

Machine check analysis and recording (MCAR) and channel check handler (CCH) routines for the Model 135 represent an extension of the recovery support provided by DOS for System/360 Models 30, 40, and 50. The MCRR routine, which provides machine check and channel error handling, is offered as an option for these models.

MCAR, CCH, and the I/O error recording routines OBR and SDR will be included automatically in any DOS supervisor generated for a Model 135. The resident processor storage requirement for these four routines is approximately 5700 bytes in the supervisor area. A Model 30 DOS user with MCRR and OBR/SDR included in the supervisor being used (4800 bytes) will experience approximately 900 bytes increase in supervisor size because of the inclusion of Model 135 RMS.

The two primary objectives of RMS are (1) to reduce the number of system terminations that result from machine malfunctions and (2) to minimize the impact of such incidents. These objectives are accomplished by programmed recovery to allow system operations to continue whenever possible and by the recording of system status for both transient (corrected) and permanent (uncorrected) hardware errors.

Machine Check Analysis and Recording

After IPL of a control program containing Model 135 RMS routines, all machine check mask bits are enabled so that all soft and hard machine checks will cause an interrupt. MCAR receives control after all machine checks.

When a System Recovery soft machine check occurs to indicate a successful instruction retry, an environment record (recovery report), containing pertinent status information from the fixed area, recovery action, program identification, date, and time of day, is constructed by MCAR and written in the environmental recording data set (ERDS), whose symbolic unit name is SYSREC (corresponding to the SYS1.LOGREC recording data set of OS). The operator is informed that a soft machine check has occurred.

If the soft error occurred during execution of nonprivileged code (a processing program, for example), error recording takes place when the interrupt occurs. If the soft error occurred during execution of privileged code (supervisor code, for example), soft machine checks are disabled, an indication that MCAR processing is required is stored, and control is returned to the interrupted privileged code. When execution of the privileged code terminates, control is given to the soft machine check handler portion of MCAR to record the soft error. Machine check statistics are lost during the interval between occurrence of the soft error and its recording if another machine check condition arises. Specifically, a hard machine check interrupt during this interval will overlay the soft machine check logout and any additional soft machine check conditions will result in the loss of all but one of the soft errors, since the Model 135 can keep only one machine check pending at a time for each machine check type.

When a System Recovery interrupt occurs to indicate that the EFL for single-bit storage error corrections has been reached, the error is recorded, a message is given to the operator, and system operation continues.

When a system recovery message is received during system operation, for either CPU or storage errors, repair should be done during the next scheduled maintenance period or can be done during a deferred maintenance period after system operations are terminated normally.

An interrupt because of an error in the time of day clock or interval timer results in error recording, a message to the operator, and continued system operation.

When an Instruction Processing Damage hard machine check occurs (uncorrectable or unretryable CPU error, double- or multiple-bit storage error, or storage protect key error) during the execution of supervisor (or any privileged) code, the system is placed in a hard wait state after an attempt is made to prepare and record a damage report record. MCAR does not attempt to refresh damaged supervisor code. (Hard machine checks that occur during an attempt to access critical data or code on SYSRES also result in system termination.) The occurrence of an Instruction Processing Damage interrupt during processing program execution always results in an attempt to terminate the task involved after the error is recorded. System operation then continues.

MCAR performs repair procedures if a storage protect key failure or double- or multiple-bit processor storage error occurs in a processing program partition. Validation of damaged processor storage is attempted by moving a doubleword of binary zeros and then ones into the area. If the storage protect key repair or storage validation procedure fails, dynamic reallocation of the partition is attempted. A routine is executed to determine in which half of the processing partition the storage error occurred. If the failure occurred in the high-order half of the partition, the upper boundary of the partition is lowered to the first 2K boundary address below the address of the error location. If the failure occurred in the low-order half of a foreground partition, the lower boundary of the foreground partition is raised to the first 2K boundary address above the address of the error location. (The lower boundary of the background partition cannot be raised.)

A message is issued to the operator to inform him of the new upper or lower boundary address and the current size of the partition whenever a successful reallocation is performed. Only programs less than or equal to the new size can then be executed in the partition, and if the lower boundary of a foreground partition has been raised, only self-relocating programs can be executed in that partition. If a foreground partition is lowered below 2K in size, or the background partition is lowered below 10K in size, or the error occurred in the lower half of the background partition, the operator is informed that the partition is no longer usable.

A System Damage hard machine check interrupt results in an attempt to record the error, followed by system termination (a hard wait state). The operator is informed of whether or not error recording was successful.

Channel Check Handler

CCH receives control after a channel error occurs. It records the error in SYSREC and passes the ECSW and other pertinent status information to the appropriate I/O error recovery routine (ERP) unless analysis of the error indicates that system operation cannot continue (the error involved SYSRES, for example). If the ERP can correct the error, operations continue. If a permanent channel error exists, CCH records the error and cancels the partition affected. The operator is notified. System termination occurs (1) if a hard channel error occurs during the access of program phases or critical data contained

on SYSRES, (2) if two channels are damaged at the same time, or (3) if more than four channel errors are outstanding concurrently.

The recovery support provided by the MCAR and CCH routines represents an extension of the facilities provided by the optional MCRR routine of DOS, which does not contain any repair or channel retry procedures.

RECOVERY MANAGEMENT SUPPORT (RMS) FOR OS MFT

RMS for the Model 135 consists of extensions to the facilities offered by RMS routines currently provided in OS for Models 65 and up. The two RMS routines, machine check handler (MCH) to handle machine check interrupts and channel check handler (CCH) to handle certain channel errors, will be included automatically in MFT control programs generated for the Model 135. Approximately 7000 bytes of resident processor storage are required for Model 135 recovery management.

Machine Check Handler

After IPL of a control program containing Model 135 RMS routines, all machine check mask bits are enabled to permit all soft and hard machine checks to cause an interrupt. MCH receives control after the occurrence of both soft and hard machine check interrupts.

When a System Recovery soft machine check occurs to indicate a successful instruction retry, MCH formats a recovery report record to be written in the system error recording data set SYS1.LOGREC. This record contains pertinent information about the error, including the data in the fixed logout area, an indication of the recovery that occurred, identification of the job and program involved in the error, the date, and the time of day. MCH schedules the writing of the recovery report record and informs the operator that a soft machine check has occurred.

When a System Recovery soft machine check occurs to indicate that the EFL for single-bit storage error corrections has been reached, MCH disables the EFL mask bit, the error is recorded, the operator is informed, and system operation continues.

If a system recovery message is received during system operation, CPU or storage repair should be done during the next maintenance period.

The operator also is informed of the occurrence of a Time of Day Clock Damage or an Interval Timer Damage machine check interrupt. Error recording is performed, after which the system is placed in a wait state if a time of day clock error occurred. System operation continues after an interval timer error.

When an Instruction Processing Damage hard machine check occurs (uncorrectable or unretryable CPU error, double- or multiple-bit processor or control storage error, or storage protect key failure), MCH attempts to identify the task associated with the error so that the task can be terminated abnormally. A damage report record, which contains both the fixed logout area data, the recovery action taken, the program and job identification, the date, and the time of day, is prepared and logged. System operation continues if the task associated with an uncorrectable error can be identified and terminated. System operation halts, and a re-IPL is required if the error involves

control storage (a double- or multiple-bit error), if an uncorrectable error damages a portion of the control program, or if the error cannot be associated with a specific task. The operator is informed of whatever action is taken.

When a System Damage hard machine check occurs, programmed recovery is not possible, and MCH places the system in a wait state after an error record logout and termination procedures are attempted.

MCH for the Model 135 contains model-dependent routines and will not execute correctly on System/360 models or another System/370 model. (See Section 60:30 for a discussion of operating system portability.)

Channel Check Handler

CCH receives control after a channel error causes an I/O interrupt. CCH formats both an error information block (containing the limited channel logout data) for use by an ERP routine and a CCH error record for recording in SYS1.LOGREC. The latter contains status information from the logout area, the ECSW, program identification, date, and time of day.

If CCH determines that operating system integrity has been impaired by the channel error, control is given to MCH for error recording, and system operations are terminated. Otherwise, the error information block and error record are passed to the appropriate device-dependent error recovery procedure (ERP), which logs the error record and retries the failing I/O operation, using status information from the error information block. If a successful retry occurs, system operation continues. If the error is deemed permanent (uncorrectable), another error record is prepared and recorded by the outboard recorder routine (OBR), and the task involved is abnormally terminated (unless I/O RMS or a user-written permanent error handling routine is present). The operator is informed of the abnormal termination and system operation continues.

The CCH routine is structured in a manner that makes it model independent. A channel/model-independent module resides in the operating system nucleus. The required channel-dependent modules for the Model 135 included in the operating system at system generation time are loaded during the IPL procedure. The nucleus initialization program (NIP), using channel configuration data specified by the user at system generation time and the new STORE CHANNEL ID instruction, determines the types of channels present in the system. OS CCH routines are, therefore, compatible for System/370 models, for System/360 Models 65 and up, and for MP/65 systems.

Figure 50.10.5 at the end of this subsection shows the general flow of programmed error recovery procedures after an I/O interrupt.

ERROR RECOVERY PROCEDURES (ERP'S) - DOS AND OS

These device-dependent error routines are a standard part of the control program generated for any DOS or OS environment. OS ERP's will be modified to accept and use limited channel logout ECSW data formatted by the CCH routine after a channel error. The ECSW provided by the DOS CCH routine will be handled by a set of completely new CCH ERP routines. The DOS CCH ERP's are an addition to the current set of DOS ERP's. The latter will be used without modification.

OS ERP and DOS ERP routines written for the 3211 Printer and the 3330 facility will include support of the larger number of sense bytes provided by the control units of these devices.

When a channel or an I/O device error occurs on a Model 135, the appropriate ERP is scheduled to perform recovery procedures. If the error is corrected, operations continue normally. If the error cannot be corrected (it is permanent), error recording occurs. If I/O RMS (for OS only) or a user-written permanent error handling routine is not present, the affected OS or DOS task is abnormally terminated. The operator is notified of permanent I/O errors. (See Figure 50.10.5.)

STATISTICAL DATA RECORDER (SDR) AND OUTBOARD RECORDER (OBR) - DOS AND OS

OBR and SDR are included in any DOS supervisor generated for a Model 135. OBR and SDR routines are included in all OS control programs.

These routines are requested by the ERP routines during their processing of error conditions. The SDR routine is requested when one of the error statistics counters becomes full. Counters are maintained in the resident control program storage area for each I/O device in the system configuration. SDR records these statistics in the appropriate SDR summary record for that device contained in the error log data set (SYSREC for DOS, SYS1.LOGREC for OS). This ensures recording of temporary I/O device error data. The OBR routine of OS records both temporary and permanent channel errors (handled by the CCH routine in DOS) and writes an outboard record containing pertinent status data whenever a permanent error occurs for a device. SDR is also executed to write accumulated statistics for that device when a permanent error occurs. (See Figure 50.10.5.)

ENVIRONMENT RECORDING, EDIT, AND PRINT PROGRAM (EREP) - DOS AND OS

The currently available EREP routine for DOS is a special purpose utility that can be initiated as a job step via job control statements in the input stream or by an operator command entered via the console. Its function is to edit and print all error records contained in the SYSREC recorder file. EREP will be extended to handle all status records written by DOS Model 135 machine and I/O error recovery routines and will be included in all DOS operating systems generated for the Model 135. Modifications to the current EREP for DOS will enable it to perform items 1 and 2 outlined below for the OS EREP routine.

OS EREP is a standard system utility that can be initiated as a job step via standard job control statements at any time. It contains model-dependent routines and will be extended to handle the status records written by System/370 OS RMS routines. It performs the following:

1. Edits and prints all error records contained in SYS1.LOGREC. These records have been constructed and/or written by machine and I/O error handling routines such as MCH, CCH, OBR, and SDR.

2. Accumulates a history of specified record types from SYS1.LOGREC by creating or updating an accumulation data set

3. Edits and prints a summary of selected records from SYS1.LOGREC or an accumulation data set

I/O RMS FOR OS

I/O RMS routines are optional, model-independent routines supported in MFT (and MVT) environments. These reconfiguration procedures attempt to minimize the number of abnormal job terminations and unscheduled system halts that occur because of errors on channels or I/O devices.

The alternate path retry (APR) routine provides for the retry of a failing I/O operation on another channel path to the device involved, if one is available, when an uncorrectable channel error occurs. Thus APR, if present, is entered from a device-dependent ERP when a permanent error is deemed to exist after retry procedures have been attempted. If the I/O error is corrected using the alternate channel path, operations continue. If a permanent error still exists, the task is abnormally terminated unless the dynamic device reconfiguration routine is present. A malfunctioning channel path can be varied offline by the operator if necessary.

The dynamic device reconfiguration (DDR) routine permits the operator to move a demountable volume from one device to another of the same type when a permanent hardware error occurs and provides repositioning of the volume so that the failing I/O operation can be retried. A volume can also be demounted so that device cleaning procedures can be performed and it can then be remounted on the same device. The DDR option also supports demountable system residence devices and unit record equipment. DDR is entered from a device-dependent ERP after a permanent channel or device error occurs on a demountable device. Task termination occurs if the error cannot be corrected and a user-written permanent error handling routine is not present. (See Figure 50.10.5.)

I/O RMS is not included in DOS support, which handles alternate channel paths only for tape unit switching and does not provide dynamic I/O device allocation by the supervisor.

CHECKPOINT/RESTART FACILITIES FOR DOS

Programs terminated because of an I/O device or channel error or as a result of a system termination can be restarted from a checkpoint or from the beginning of the job step if their job control is resubmitted with the appropriate restart control statements included. Malfunctioning I/O devices can be removed from the table of available devices by the operator, and different devices of the same type can be assigned to job steps via their job control or by the operator. Warm start facilities are not required, since DOS does not build work queues. (DOS POWER, which builds input and output queues, does provide a warm start facility.) DOS users should plan program and system restart procedures.

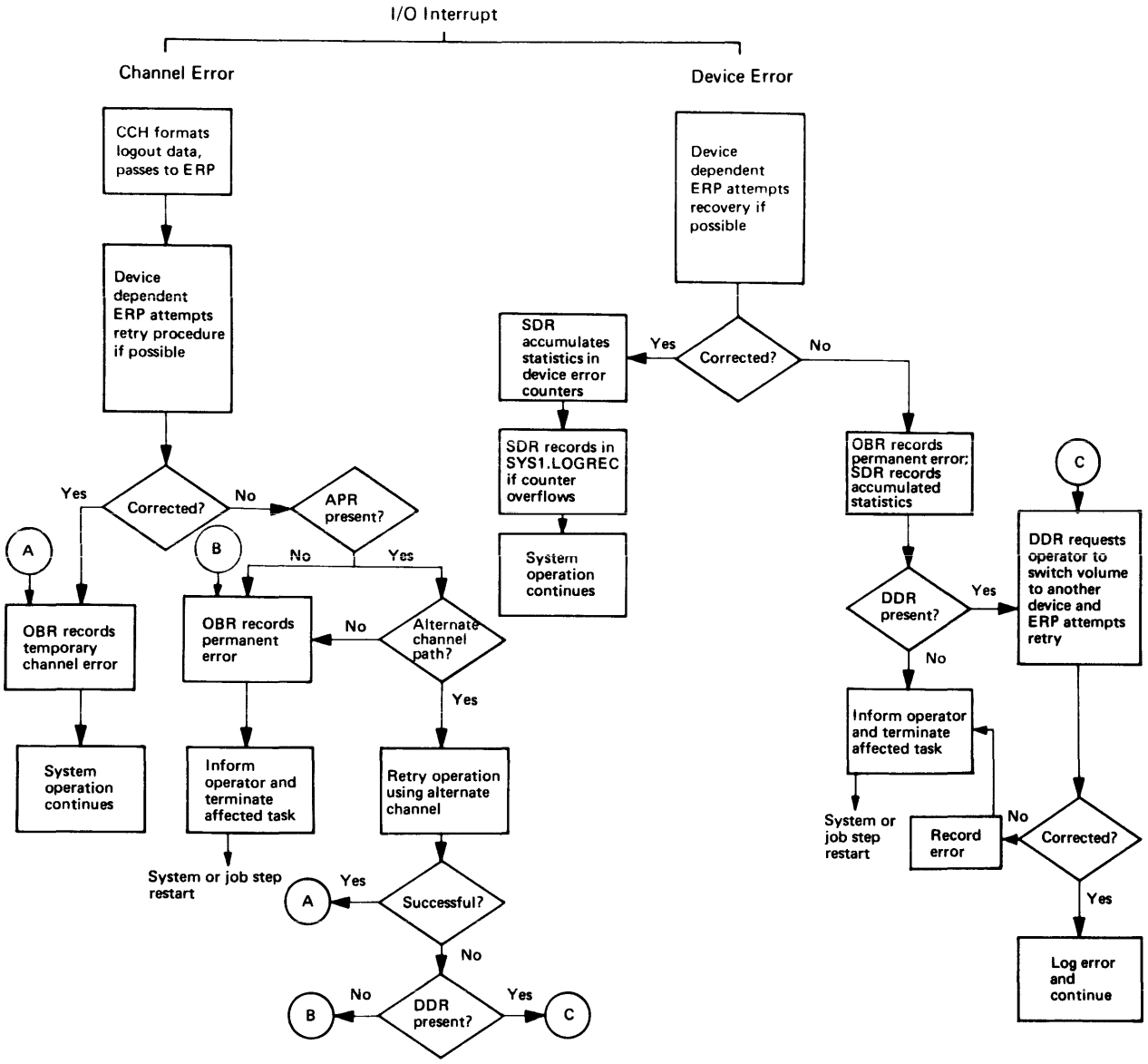


Figure 50.10.5. General flow of OS error recovery procedures after an I/O interrupt

ADVANCED CHECKPOINT/RESTART AND WARM START FACILITIES FOR OS

If the RMS and I/O RMS routines fail in their attempt to correct a hardware error and the error is one that causes program or system termination, the automatically provided advanced checkpoint/restart and warm start facilities of OS can be employed to minimize the impact of the termination on system operation. The automatic restart facility can be used to cause terminated programs to be rescheduled immediately without resubmission of their job control, so that a minimum of operator intervention is required. The operator must authorize all automatic job step restarts. If a permanent I/O device or channel failure caused the program termination, the device or channel can be varied offline. This will ensure allocation of a different device when the program step is reinitiated.

The warm start facilities of the control program provide automatic saving of SYSIN and SYSOUT data sets and input and output work queues so that processed work is not lost when a system termination occurs. The operator is informed of the status of jobs in execution when the system terminated, and these jobs should be restarted automatically from the beginning or from a checkpoint if the type of processing involved permits such a restart. System design should include planned restart procedures for unscheduled terminations of individual programs and the system. (See Figure 50.10.5.)

50:15 REPAIR FEATURES

The programmed repair features supplied are designed to minimize the impact of malfunction diagnosis and repair on system availability. Fault location and repair time should be reduced by:

1. Improved error recording. Both intermittent and solid hardware and I/O errors are logged at the time of failure if Model 135 RMS routines are present. More status information will be available than was recorded previously for Models 25 and 30.

2. Online error diagnosis. Error diagnosis and repair can be performed concurrently with system operation in a multiprogramming environment when necessary to avoid total system, direct access facility, or tape subsystem unavailability.

3. User execution of diagnostic routines. More exacting diagnostic routines are available for execution by the console operator.

The following maintenance and diagnostic routines will be provided:

- Online Test Executive program (OLTEP) and Online Tests (OLT's) for operation under DOS and OS control to test malfunctioning I/O devices concurrently with system operation

- Diagnostics contained on disk cartridges for use in diagnosing malfunctioning field-replaceable units (stand-alone routines)

OLTEP AND OLT'S - DOS AND OS

OLTEP is designed to operate as a processing program under operating system control. It handles the required interface between the operating system and the device-dependent OLT's. One OLTEP will be provided for operation under DOS and another for execution under OS. These two OLTEP's support functions not provided by the currently available DOS OLTEP and OS OLTEP programs for System/360.

The inclusion of OLTEP in an operating system is automatic for a DOS control program generated for a Model 135. (The minimum DOS supervisor size for a Model 135 includes the resident OLTEP storage requirement.) OLTEP is a system generation option for OS users. (The size of the resident OS control program is not increased by the inclusion of OLTEP.) A stand-alone version of OLTEP, called OLTSEP, will be available as well.

OLTEP directs the selection, loading, and execution of device-dependent OLT's for the purpose of I/O device testing and error diagnosis. OLTEP is also designed to verify I/O device repairs and engineering changes. OLTEP and OLT's will be used by the customer engineer.

As with any other job step, OS OLTEP is invoked with job control and executes with a user-assigned priority. A minimum program area

of 28K is required for OLTEP operation in OS environments. DOS OLTEP, also invoked via job control statements, operates only in the background partition in a minimum of 14K and cannot be run with a 6K supervisor. The input stream or system console device can be used to supply the parameters required for test operations (devices to be tested, options desired, etc.). Both OS OLTEP and DOS OLTEP ensure the protection and security of user data files and storage in use while OLT's are operating. OS OLTEP also ensures that the devices to be tested are online or offline (as far as the operating system is concerned) as required by the particular device type.

The OLTEP's for OS and DOS also have the new capability of being able to access history records describing previous I/O errors on the device being tested. In addition, multiple devices can be tested during one OLTEP execution. If a console is used to define the test run, the new prompting facility can be requested as an aid to the user supplying the definition.

OLTEP and the OLT's will reside in a DOS core image library. In OS environments portions of OLTEP will reside in both SYS1.LINKLIB and SYS1.SVCLIB, while the OLT's can be placed in a user-designated disk library (partitioned data set).

OLTEP and OLT's can operate concurrently with other executing jobs in a multiprogramming environment and provide online I/O device testing and repair, eliminating the necessity for complete system unavailability while many types of errors are being diagnosed.

MAINTENANCE AND DIAGNOSTICS

The diagnostic programs and maintenance procedures for the Model 135, in conjunction with the system's hardware and programmed error recovery and recording, are designed to result in increased system availability through a combination of predictive maintenance and preventive maintenance. While system availability is of greatest importance to installations with online operations, most installations have a time period in their daily data processing during which system availability is critical.

Predictive maintenance for the Model 135 involves using a set of uncomplicated procedures that can readily be incorporated into the normal operating procedures handled by the system console operator. These procedures consist primarily of operator execution of IBM-supplied diagnostic programs that are designed to (1) identify intermittent failures, which if not repaired generally become solid and cause system stops, and (2) isolate the error to the failing unit with identification of the failing part when possible.

The objective of predictive maintenance procedures, increased system availability, is to be accomplished by attempting to:

- Reduce the number of unscheduled system interrupts. Identification of system degradation before system failure permits repair on a scheduled or deferred maintenance basis.

- Eliminate system down time because of "no trouble found" situations. A diagnostic routine (ASCP) is provided that can be used in many instances to differentiate between hardware and nonhardware errors.

Each installation receives five console disk cartridges for a Model 135 system. One contains microdiagnostics, in addition to the system microcode. A copy of this disk cartridge, the initial microprogram (IMP) disk, is provided as backup. The system test program, called the Automatic System Checkout Program (ASCP), is contained on two disks and the Device Diagnostics Program (DDP), for checkout of certain integrated adapters, is provided on one disk. These disks are clearly labeled and color-coded by function for easy identification. The color and labeling on a disk cartridge mounted in the console file are visible through the console window to show which disk is mounted and whether it is mounted right-side up.

Initial Microprogram (IMP) disk. The CPU microdiagnostics on this disk are automatically executed every time an initial microprogram load is performed. They execute directly from the console file prior to the loading of control storage with system microcode and are designed to check out basic system hardware before operations begin. These microdiagnostics detect most possible faults in the CPU, storage, and the channels as well as the majority of possible faults in the Integrated Communications Adapter. The microdiagnostics required to test all system features are contained on all IMP disks. Each feature-dependent routine determines, prior to executing, whether or not the feature is present for the system.

Execution of the microdiagnostics and loading of the microprogram require approximately one minute. If an error is detected during execution of IMP microdiagnostics, the system stops, with error indications displayed on the system control panel. Errors reported during IMPL should be repaired before normal system operation is attempted.

Automatic System Checkout Program (ASCP). This program, contained on two disk cartridges, quickly checks out most of the components of the system. ASCP identifies failing I/O units and certain CPU errors.

ASCP executes as a stand-alone program in processor storage and thus control storage must already contain the system microcode. Once ASCP has been loaded from the console file, it begins execution and lists the I/O devices to be tested, which are all those that were in ready status when ASCP execution began. After operator verification of the devices to be tested, I/O tests on all these devices are initiated and they run concurrently with CPU tests to provide maximum system use and thus checkout.

The output from one complete ASCP test is a reliability report indicating CPU and I/O errors, if any. A printer or the console printer-keyboard can be used for output. The complete set of tests continues to be executed until terminated by the operator and output reports are sequentially numbered. Repetition of the tests increases the reliability of the output. When an error is detected, the data from the report should be reported to IBM. Depending on the results of ASCP, immediate or deferred maintenance is necessary. An example of a situation in which deferred maintenance is possible is when ASCP identifies a malfunctioning I/O device. If jobs can be run without the use of the malfunctioning I/O device, maintenance can be deferred. (In an OS environment, the device can be varied offline, while in a DOS environment the symbolic device address of the malfunctioning I/O

device can be reassigned to another device of the same type.) If the error on an I/O device cannot be identified as a result of running ASCP, the CE can run the device-dependent OLT's for the I/O unit, online, under OLTEP, or in a stand-alone environment using OLTSEP. ASCP or OLT(S)EP can then be used to verify repair of the failure.

ASCP will be of the most benefit if it is run by the operator at the following times:

1. During system initiation every day, or whenever system power is turned on, to check out system operation

2. When hard errors occur that terminate system operation (prior to contacting the customer engineer)

3. Whenever errors occur that cannot be identified as hardware or nonhardware. If ASCP executes without detecting an error, the possibility of a programming or operational error should be investigated.

The recommended operator procedure for initiating system operation after a power-on is to follow the initial microprogram load procedure with execution of ASCP. In general, at least two reports should be generated. Run time for ASCP (excluding setup) for a medium system (with a reader, a punch, a printer, and channel attached disk storage) is 4 minutes plus 10 seconds per report. Run time for a large system (a reader, a punch, two printers, teleprocessing, 6 tape units, and 3330 disk storage) is 6 minutes plus 45 seconds per report. Once ASCP has run successfully, the operating system can be IPLed.

Device Diagnostics Program (DDP) disk. This disk contains time-dependent microdiagnostics for the integrated file adapter and the integrated console adapters. They are designed to be executed by the customer engineer to locate faults not identified by the CPU microdiagnostics. Execution of microdiagnostics contained on the DDP disks detects most possible faults in the integrated file adapter.

50:20 RAS SUMMARY

The Model 135 offers significantly more hardware and programming systems RAS features than are provided for Models 25 and 30. These features enhance the Model 135's capability of performing successfully in the environment required by existing and new data processing applications. The greatest benefit from RAS features can be obtained when an installation uses all the RAS facilities available. It is desirable for Model 135 users to design a system that includes RAS features and to become involved in the implementation and use of maintenance procedures and aids. Specifically, the user can:

- Install a version of DOS or OS MFT that includes Model 135 support so that RAS features are included in the operating system generated

- Have the operator use IBM-supplied diagnostic programs as suggested

- Include OLTEP and OLT's in his operating system (optional for OS users)

- Plan system and program recovery procedures (use of checkpoint/restart and warm start facilities)

- Have operating personnel perform normal hardware maintenance procedures, such as the periodic cleaning of tape unit heads. Proper system hygiene should be maintained, in general.

- Implement an effective program of operator training so that the number of system malfunctions that occur because of operator error is minimized

Although console operator involvement in the execution of diagnostic programs is not mandatory, operating procedures and documentation for these programs have been designed for operator use so that a minimum of effort is required to train operators to run them. Human factors have been taken into account in the design of the system control panel, the error logout sheet, disk cartridge identification, and actual operating procedures. The latter will be explicitly outlined in the operator's guide for the Model 135.

The benefits to an installation, in increased system availability, that can result from user participation in the maintenance procedures outlined should make the operator time involved more than worthwhile.

60:05 GENERAL CONSIDERATIONS

The hardware, I/O devices, and programming systems support offered by the Model 135 make it an attractive system for both DOS and OS MFT users. The Model 135 provides DOS and OS users with the potential for increasing the throughput of existing applications and the capability of expanding into new application areas on an improved price performance basis.

Specifically, the Model 135 offers Model 25 and Model 30 DOS users:

- Increased internal performance, approximately two to seven times the Model 30 and three and one-half to sixteen times the Model 25
- Lower cost entry into terminal-oriented applications using the ICA and 2319 disk storage attached via the IFA
- Additional channel capability--an IFA plus two selector channels--and faster channel data rates
- The capability of attaching more and faster I/O devices. For the Model 30 user, the Model 135 offers attachment of more 2314 disk storage, 3330 facilities, and 320 KB tape units. For the Model 25 user, the Model 135 offers attachment of 2314 and/or 3330 disk storage, magnetic tapes faster than 60 KB (up to 320 KB), and the 3211 Printer.
- The ability to attach 3505 readers and 3525 punches with new features that offer expanded capabilities for card processing
- Integrated 1400 emulation that provides advantages over and above those offered by CS/30.
- Improved throughput potential and additional functions through use of more processor storage, increments of which can be added for less cost than on Models 25 and 30

In addition to the general uses stated in Section 1, additional processor storage can be used in a DOS environment for the following:

- Installation of POWER II to provide unit record data transcription (card reader, printer, and punch operations) overlapped with production job step processing. DOS users with a version of POWER already installed can allocate more or larger I/O areas and handle more devices to improve performance. The remote job entry function of POWER II can be used also. A minimum partition size of 38K is required to support remote job entry in addition to data transcription.
- Execution of full-function language translators such as ANS FULL COBOL V3 (54K), the PL/I Optimizing Compiler that has OS PL/I F capability (44K), and FORTRAN IV (40K)
- Execution of the Tape and Disk Sort/Merge program, which offers significantly improved performance when larger amounts of processor storage are used
- Installation of time sharing, using the Interactive Terminal Facility (ITF)--a partition size of approximately 40K is required to handle 10 to 12 terminals

- Implementation of DATA/360-DOS for data entry and verification operations, using 2260 Display Stations instead of keypunches and verifiers

- Installation of GPSS (General Purpose Simulation System)

- Installation of IBM-supplied application-oriented programs (Type II and program product)

Model 135 DOS users can also take advantage of new DOS facilities such as:

- Multiple core image libraries for executable program (program phase) residence instead of a maximum of one. Each batched partition in the system can have a private core image library assigned.

- The ability to compile, link-edit, and execute certain system service programs in batched foreground partitions. Previously, these functions could be performed only in the background partition. Most IBM-supplied language translators (Type I and program product) can be executed in a batched foreground partition, as can CSERV, MAINT, and DSERV service programs.

- Data file protection at open time to prevent unauthorized access to data by programs

- Problem determination aids (dumps, traces, etc.) that are designed to identify a system failure as one of three types--a system hardware error, a system programming error, or a user error--so that the failure can be pinpointed and corrected more quickly.

DOS users who find it desirable to convert to OS with installation of a Model 135 will find the transition eased by use of DOS emulation under OS, a feature not available on System/360 models. For larger DOS users, the following are some of the attractive features offered by OS MFT that are not provided by DOS:

- Expanded multiprogramming--up to 15 processing programs (partitions) operating concurrently with unit record data transcription (multiple reader interpreters and output writers)

- Priority and job class job scheduling for more efficient use of available system resources

- Dynamic resource allocation (at program execution time) by the operating system of I/O devices, direct access space, processor storage, and programs

- Expanded data management facilities, including device independence, a data set catalog, shared direct access device support, and graphic device support

- Program relocation at program load time by the control program

- Extensive job accounting data and resource usage statistics gathered by the control program (significantly more than are provided in DOS)

- Expanded teleprocessing support (TCAM) and time-sharing support (TSO and CALL/360-OS)

- Alternate and multiple console support, including graphic consoles

In addition to supporting a wide variety of general and specific application programs, OS MFT also supports Model 135 features such as extended precision, the time of day clock, rotational position sensing on 3330 facilities, and block multiplexing. The Model 30 OS PCP user can now use integrated instead of stand-alone 1400 emulation and, like the DOS user, will benefit from the expanded hardware capabilities --faster internal performance, more and faster channels and I/O devices, larger processor storage, etc.--offered by the Model 135.

Because extensive hardware and programming systems compatibility exists between the System/370 Model 135 and System/360 models, most Model 25 and 30 users can upgrade to a Model 135 with a minimum of effort. Essentially, no more effort may be involved in the Model 135 installation process for OS PCP users and DOS users who do not convert to OS than is required currently to change from one DOS release to another, to change from OS PCP to OS MFT, or to regenerate an operating system to include new hardware, new I/O devices, and more control program options. In most cases, the fact that an OS PCP or DOS user is upgrading to a Model 135 should not add to the effort that would be required if new applications were to be added and system changes were to be implemented for a Model 25 or 30 upgrade to another System/360.

60:10 DOS TRANSITION - MODELS 25 AND 30

A system generation must be performed using a release of DOS that includes Model 135 support in order to obtain a supervisor that supports new Model 135 features. The DOS starter system can be used on a System/360 or System/370 to generate a control program for either a System/370 or a System/360 model. The existing system generation job stream can be used, modified to reflect the Model 135 hardware configuration and the use of integrated emulator programs, as appropriate. Additional supervisor options can be selected as well.

A DOS operating system generated for the Model 135 includes the following:

- MCAR and CCH routines to handle the expanded machine check interrupt. OBR and SDR are included to handle I/O error recording, and OLTEP is present also.
- Support of the new I/O devices specified (3211 Printer, 3505 Card Reader, 3525 Card Punch, 2319 and 3330 disk storage, and 3803/3420 tape subsystem) and the new Model 135 console indicated. Support of the ICA is provided also if teleprocessing support is requested.
- Support of the interval timer, if requested
- Support of the new instructions by Assembler D (14K variant)
- The required interface to the integrated emulator program specified at system generation, if any

In general, the new supervisor will be larger than the one currently in use because of the automatic inclusion of MCAR, CCH, OBR, SDR, and OLTEP routines and user selection of additional options. (The minimum DOS supervisor size for the Model 135 is 12K.) This increase will be less for DOS supervisors that currently contain the optional MCRR, OBR, and SDR routines. A larger supervisor and the availability of additional processor storage in the Model 135 will cause partition starting addresses to change. Therefore, existing user-written, nonrelocatable DOS programs have to be link-edited relative to the new partition starting addresses and placed in the new core image library. If relocatable modules for these nonrelocatable programs

are not available, reassembly of the source modules, as well as link editing of the resulting relocatable modules, is required. Existing user-written, self-relocatable DOS program phases can be copied directly from the old core image library to the new one.

Subject to the restrictions stated in Section 10:15, alteration of an existing source program is required only if new processing is added, if existing processing is changed, if there is a change in the I/O device types used in the program (ASSGN job control statements may also require changes), or if the program contains a user-written routine that depends upon a particular release of DOS for communication with the supervisor. Subject to the same restrictions, programs written for a 2701 can be executed on a Model 135 with the ICA and/or a 2701 (with comparable features) without modification. BTAM and QTAM programs written for a 2702, 2703, or Model 25 ICA can be executed on a Model 135 with a 2701, 2702, or 2703 or the ICA with appropriate features if the operating system has been regenerated.

Existing printer programs that specify the 1403 Printer in the DTF (either by default or specifically) can be used without change for a 3211 Printer to support the same functions as before. Additional job steps that load the FCB and the UCB must be added to the input stream where appropriate. Source programs must be modified if support of any of the new 3211 features (retry after error, etc.) is desired or if a function or feature that was not used previously is to be supported on the 3211.

Conversion from 2311's (or 2321's) to a 2319 or 2314 facility involves transfer of the data to the new devices, and program and job control statement modifications. (The same procedures are required for conversion from 2314's or 2321's to a 3330.) The amount of program alteration required depends on the file organization used and changes made to take advantage of the larger capacity of the new disk pack (larger blocking factors, changes in ISAM area allocations, etc.). In all cases, the DTF in the program must be modified to specify the new direct access device, and if block size is changed, I/O areas must be redefined. Device assignments and extents specified in the job control statements for these programs must be changed also.

Sequentially organized files on 2311's or 2321's (processed by SAM) can be copied directly from the source direct access device to the new device, using the Disk-to-Disk or Data Cell-to-Disk utility. Alternately, files can be dumped to tape and then loaded on the new disk pack, using disk-to-tape and tape-to-disk utilities if the source direct access device is not in the Model 135 configuration.

Indexed sequential and directly organized files (processed using ISAM and DAM, respectively) must be transferred to the new disk pack with a user-written program. The original creation or reorganization program with appropriate modifications can probably be used. The randomizing routine used in programs that process directly organized files must be modified if relative track or actual track address record reference is used and fewer or more tracks are allocated to the file than before.

User-written programs that use the EXCP macro to access files that have been transferred to new disk packs may have to be modified to reflect the characteristics of the file on the new pack, a different number of records per track, a different number of tracks per cylinder, etc. All 2311 CCW lists will operate on 2314/2319 disk drives except those that are device or channel time dependent. CCW lists for the 2321 will operate also, except when they contain a device-dependent command such as RESTORE. (All 2314 CCW lists will operate on 3330 facilities except those that are device or channel time dependent and

those that support the file scan feature, which is not available on the 3330.)

As stated in Section 20:25, existing tape processing programs and their job control statements and tape volumes need not be modified in order to be used with 3803/3420 subsystems with equivalent features whenever the same recording modes are used.

Whenever possible, seven- and nine-track NRZI mode tape volumes should be converted to 1600-BPI PE format to obtain the benefits of the higher density and the PE recording technique. In cases in which tape volumes must retain seven-track NRZI format, for interchange with other systems, for example, use of the 3803/3420 subsystem offers improved tape reliability and subsystem serviceability, as already discussed.

Conversion of seven- and nine-track NRZI tape volumes can be done gradually during production processing. That is, the old master input tape volume is read in on a 3420 tape unit with the appropriate compatibility (Dual Density or Seven-Track) feature, while the new master output tape is written on a 3420 tape unit in 1600-BPI PE format. Existing programs that process these converted tapes need not be modified unless a DTF parameter, such as blocking factor, is altered. Existing job control for these programs (ASSGN statements) must be changed to reflect the new recording mode.

Tapes that cannot be converted on an as-used basis, such as program tapes or active reference tapes that are not rewritten when processed, can be converted using a copy utility.

Existing user-written Assembler and high-level language programs for the 2540, 2501, 2520, and 1442, with the exception of RPG II, can be executed without modification using a 3505 reader or 3525 punch with comparable features. RPG II programs must be modified to specify the 3505 or 3525 and reassembled. Assembler and high-level language programs using a combined read/punch file, except those written in RPG II, must be modified to define two separate files. RPG II programs with combined file operations do not require modification. In all cases, job control ASSGN statements for both reader and punch programs must be altered to allocate a 3505 or a 3525, as appropriate.

60:15 DOS PORTABILITY

A DOS user with multiple Model 135 systems can generate a single operating system that can be used on each system, subject to restraints imposed by differences in hardware configurations and engineering change levels.

A single DOS control program can also be portable among similarly configured System/370 Models 135, 145, and 155 because of the way DOS RMS routines are structured. (DOS support is not provided for the System/370 Model 165.) During IPL, the determination of the model in use is made by the initialization routine, using the STORE CPU ID instruction. Bits are then set in the supervisor that are tested during RMS routine execution to determine which model routine should be used in those cases in which model-dependent routines exist. (Note that when a Model 135 supervisor is used on a Model 145 or 155, CPU extended logout area data for these models is not recorded.)

If a single control program is to be used for both a System/370 Model 135 and a System/360 model, say 25 or 30, Model 135 should be specified at system generation time so that MCAR and CCH are included. Model 135 MCAR and CCH routines will not execute properly on a Model 25 or 30. When a DOS supervisor containing them is loaded during IPL,

MCAR and CCH are disabled automatically by the initialization routine when the routine determines that it is not operating on a Model 135. (Note that MCAR and CCH cannot be disabled by the operator with the RF parameter when a DOS supervisor containing them operates on a System/370.) This means that a Model 25 or 30 will run as it would if the optional MCRR routine was not included in the supervisor. That is, there will not be any machine check or channel error recording and the system will go into a wait state when a machine check interrupt occurs. Thus, if the facilities provided by the MCRR routine are desired for a Model 25 or 30 supervisor, a separate system generation must be done for each unique model in the installation so that the appropriate model-dependent machine check routine is included in each supervisor.

Processing programs that are to execute on the Model 135 and a System/360 or another System/370 model must use only those instructions, features, and I/O devices that are common to both systems. For example, a Model 135 program that uses byte boundary alignment or the new general purpose instructions cannot be executed on a Model 25 or 30. In addition, integrated 1400 emulator programs are not always portable among all System/370 models (see section 40:05). (Note that program products are portable from one system to another only under special licensing agreements.)

60:20 OS PCP TRANSITION

A system generation must be performed using an OS release that includes Model 135 support in order to obtain an operating system designed to support new Model 135 features. The OS starter system can be used on a System/360 or a System/370, and a control program for either a System/370 or a System/360 model can be generated.

Since Model 135 features are not supported by OS PCP, conversion from PCP to MFT is required. This involves redesign of the system to establish the number, size, and types of partitions to be defined. The MFT options to be included in the control program must also be determined. Installation-written routines that interface with the control program, such as an accounting routine, must be inspected and modified where necessary to interface with MFT instead of PCP.

The existing system generation job stream can be used as a base for the new system generation input required to generate an MFT control program. It must be modified where appropriate, as indicated below.

- Stage I input must be altered to include the parameters required to generate an MFT instead of a PCP control program. It must also be modified to reflect the Model 135 configuration, including the presence of any new I/O devices or features, such as the 3505 Card Reader, the 3525 Card Punch, the 3211 Printer, the 3330 facility, integrated emulator programs, etc. Other control program options, such as I/O RMS, OLTEP, and performance improvement features, can be included also.

- Direct access space allocation for operating system data sets will have to be adjusted as indicated in IBM System/360 Operating System, System Generation (GC28-6554). If the system residence device type is to be changed, to a 2314 or a 3330, for example, UNIT parameters in job control statements must be changed where necessary.

- FCB and UCB images should be added to SYS1.SVCLIB if a 3211 Printer is included in the configuration. User-written output writer procedures should be modified to include these specifications.

An OS MFT operating system generated for the Model 135 includes the following:

- A nucleus designed to operate in the fixed processor storage area of the Model 135. MCH, CCH, OBR, and SDR routines are included.
- Control program support of block multiplexing and rotational position sensing, as discussed in Section 30, if requested
- Support of user-specified new I/O devices and the Model 135 operator console indicated
- Support of the interval timer and time of day clock
- Support of the new instructions by Assembler F
- The required interface to the integrated emulator program specified at system generation, if any

If integrated emulator programs are to be used, the steps outlined in Section 40 must be taken in order to convert from a 1400 system or from current stand-alone emulation procedures.

EXISTING OS PCP PROCESSING PROGRAMS AND JOB CONTROL

IBM-supplied Type I OS processing programs and OS non-application oriented program products (language translators, utilities, etc.) will run on the System/370 Model 135 without alteration. Subject to the exceptions stated in Section 10:05, user-written OS processing programs that operate on the Model 30 will also execute correctly on the Model 135. Modification and reassembly of existing user-written OS processing programs are not required unless new processing is to be added or existing processing is to be altered. For example, use of an output writer in an MFT environment instead of direct printing or punching involves program alteration if the DCB used is not large enough for a direct access device and/or if device-dependent macros have been used.

Job control (JOB and EXEC) statements for all user-written programs must be modified to include required MFT parameters, such as job class, etc., and to include any desired optional parameters. Modification of the DD job control statements for these processing programs is required if I/O device type is changed (from 1403 to 3211, for example), if direct access space allocation changes, if a DCB parameter is to be altered, etc. I/O device type changes do not necessitate processing program alterations unless device-dependent macros have been used, data organization is changed, or a DCB parameter that is specified in the program is to be changed. If a procedure library is used, it will have to be recreated to include all the new job control statements.

CONVERSION TO THE 3330 FACILITY

Conversion from currently installed direct access devices to the 3330 facility involves the same procedures as are required now to change from one disk device to another, say from 2311s to 2314 disk storage. Existing disk data sets can be placed on 3336 Disk Packs, using an IBM-supplied utility in most cases. Assuming that data organization is not changed, consideration should be given to altering the block size used and the amount of space allocated to the data set. The location and size of each type area in an ISAM data set should be altered, taking into account 3336 Disk Pack characteristics. These changes can be made in job control statements.

Sequentially organized data sets (processed by QSAM or BSAM) and partitioned data sets can be copied from the source direct access device directly to the 3330 facility, using the OS IEHMOVE utility. Or they can be copied to tape and then to the 3330 facility, using the same utility (if the source direct access device type is not present in the Model 135 configuration).

Indexed sequential data sets can be copied directly from the source direct access device to the 3330 facility, using the IEBISAM utility. Alternately, they can be unloaded to tape and then reloaded, using the same utility. Changes to space allocation, etc., can be made via job control statements.

Direct organization (BDAM) data sets can be copied on a track-to-track basis from the source direct access device to the 3330 facility, using IEHMOVE, or copied to tape and then to the 3330 facility. If more records are to be placed on a 3330 track than are on a source device track, the existing reorganization program can be used to transfer the data to the 3330 facility, and the program may have to be changed. Reprogramming of the randomizing routine used in the reorganization, and in all processing programs that access the BDAM data set, is necessary if a relative track or actual address reference is used and fewer (or more) 3330 tracks are allocated to the data set than before.

Subject to the restrictions stated in Section 10:05 and those indicated for BDAM data sets, existing executable processing programs can be used without change to process data sets that have been transferred to 3336 packs. Nothing need be done to job control for these programs if the cataloged procedures supplied with the language translators are used, as long as the 3330 facility is specified as a SYSDA device at system generation. Otherwise, job control statements must be changed to request 3330 devices and, optionally, any data set characteristic changes, such as block size. RPS support, as described previously, is provided automatically.

User-written programs that use the EXCP macro to access disk data sets that have been transferred to 3336 packs may have to be modified to reflect the characteristics of the data set on the 3336, a different number of records per track, a different number of tracks per cylinder, etc. All 2311 and 2314 CCW lists will operate on 3330 facilities except those that are device or channel time dependent and those that support the file scan feature, which is not available on the 3330 facility. User-written 2311 or 2314 error routines will not execute correctly and must be modified. RPS commands have to be added by the user if this support is desired for programs that use EXCP. (Note that the XDAP macro will include support of RPS commands.)

The above discussion also applies to conversion of 2311s or 2321s to 2314A disk storage, whether the latter is channel or natively attached.

CONVERSION TO THE 3803/3420 MAGNETIC TAPE SUBSYSTEM

The DOS discussion of conversion of existing tape programs and their job control statements and tape volumes for use with a 3803/3420 tape subsystem applies to OS also except for the following. Existing programs that process tapes converted to a new density need not be modified unless an altered characteristic (recording mode or block size, for example) is specified in the program DCB. Existing job control for these programs must be altered to request a tape unit with the new recording characteristics and, if desired, to change existing DCB parameters such as block size, number of buffers, etc.

CONVERSION TO THE 3505 CARD READER AND 3525 CARD PUNCH

Existing user-written Assembler and high-level language executable programs for 2540, 2501, 2520, and 1442 readers and punches can be executed without change using a 3505 reader or a 3525 punch with comparable features. Programs that handle combined read/punch operations must be modified. This involves alteration of punch CCW lists if the EXCP macro is still to be used. If the newly defined access method support of combined operations is to be used, EXCP code must be removed, separate data sets must be defined, and the appropriate BSAM or QSAM macros must be added to replace the EXCP macro. Card data set DD statements that specifically request allocation of a reader other than the 3505 (UNIT=2540, for example) or a punch other than the 3525 must be modified.

60:25 OS PORTABILITY

To avoid multiple system generations, an OS user with multiple Model 135 systems may wish to generate a single operating system that can be used on every Model 135 in the installation. This is possible under the same system hardware and I/O device configuration restraints that exist for System/360 models. During the IPL procedure, channels and I/O devices may have to be varied offline, partition sizes may have to be redefined, etc., when the operating system is used with a different configuration than was specified during system generation.

A user with both a System/370 Model 135 and a System/370 Model 145 or a System/360 model in an installation may also wish to generate one operating system that can be used on both models. This approach provides backup when one system is unavailable and can eliminate the necessity of multiple generations.

The system generation procedure will be modified to allow generation of an operating system that is portable among System/370 Models 135 and up. Specifically, the SECMODS system generation macro will be changed to cause the inclusion of all the required model-dependent routines (such as MCH, EREP, etc.) for both the primary CPU and each secondary CPU indicated by the user. When a System/370 model is the primary model, MCH can be specified in the SECMODS macro as the error recovery routine for each secondary model. At IPL, the appropriate model-dependent routines are initialized, based on the CPU in use.

Currently, only an SER routine can be specified for secondary CPU's and this support will continue for System/360 models. Hence an OS operating system can be generated to include MCH routines for different System/370 CPU's or SER routines for different System/360 CPU's; however, one control program cannot contain a mixture of MCH and SER routines (MCH for the Model 135 and SER for the Model 40, for example).

Portability of an OS MFT operating system between a Model 135 and a System/360 Model 40 (OS MFT cannot be executed on a Model 30 since a 128K minimum is required) can be achieved by utilizing a multiple nucleus control program under the following general conditions:

1. The system hardware and I/O device configuration of both systems must be similar. For example, a Model 135 OS MFT control program generated to support block multiplexing mode and RPS direct access devices cannot be executed on a system without such channels and devices.

2. Consideration should be given to the processor storage sizes of the two models when determining the size of the scheduler, language translators, and the linkage editor(s) included in the generated system.

3. Processing programs that are to run on both models must use instructions and features common to both systems. For example, an Assembler Language program that uses the new general purpose instructions for the Model 135 or byte orientation can be executed on a Model 145 but not on a Model 40.

In order to generate an operating system that is portable between the Model 135 and the Model 40, the following steps are required:

1. A complete system generation must be performed to generate an operating system for the Model 135. The IPL-time system/operator communication option must be requested so that options specified can be altered during IPL.

2. A nucleus generation should then be done for the alternate system. The model number specified (in the SUPRVSOR, SECMODS, CENPROCS macros, etc.) will be that of the alternate system, not the Model 135.

3. Additional link edits must be performed to add model-dependent routines to the generated multiple-nucleus operating system. Specifically, SER and EREP model-dependent routines for the secondary system must be included, as appropriate.

4. If extended precision floating-point divide is used in processing programs, the following steps should be taken. SYS1.LINKLIB contains two divide simulation routines that are part of the Extended Precision Floating Point Simulator. One uses extended precision hardware, the other does not. When a full generation is performed for the Model 135, a calling mechanism is set to request the divide routine that uses extended precision instructions at execution time, since the Model 135 contains these instructions.

   Therefore, the divide simulation routine that does not use extended precision should be transferred from SYS1.LINKLIB to another library and given the same member name as the divide routine with extended precision instructions. When the operating system is executed on a Model 135, SYS1.LINKLIB should be used by extended precision programs. When the operating system is executed on a Model 40 or 50, the alternate library should be used.

Whenever a new program that is to execute on both systems is added to a library or if the Model 135 hardware configuration changes, the user must consider whether or not portability is affected. (Note that program products are portable from one system to another only under special licensing agreements.)

60:30 USE OF OTHER PROGRAMMING SYSTEMS

Subject to the restrictions stated in Section 10:05, users of OS PCP, TOS, BOS, BPS, non-IBM-supplied control programs, or OS MFT and DOS control programs not generated for a Model 135 can execute their existing control and processing programs on a Model 135 with a hardware and I/O device configuration comparable to that of the System/360 model now installed. However, since Model 135 RMS (machine check and channel check) routines are not included in these control programs, the Model 135 will operate under the following conditions:

1. Single-bit processor and control storage error correction is performed; however, a machine check interrupt does not occur after the EFL is exceeded, since the recovery mask bit is disabled. (The EFL mask bit also disabled.)

2. The IPL setting of the recovery mask bit also disables interrupts after instruction retry corrections.

3. Damage to the time of day clock or the interval timer causes a machine check interrupt condition and generation of logout data.

4. A hard machine check error (an unretryable or uncorrectable CPU error, a double- or multiple-bit processor or control storage error, or a storage protection failure) causes a hard machine check condition and generation of logout data.

A machine check control switch, which determines the action that is to be taken when a machine check condition occurs, is present on the system console. When the switch is in the process position, machine checks cause an interrupt and logout if they are enabled. This setting is to be used when an operating system containing Model 135 RMS is in operation. When the switch is in the stop after log position, all enabled machine check conditions cause an interrupt and a logout, after which the system stops.

If the Model 135 is not set to stop after log for machine checks when an operating system without Model 135 RMS is used, machine check mask bit settings are established at IPL, as described above. The system, operating in process mode, takes whatever action was planned for machine checks:

- Any control program without a recovery routine included (for example, MCRR for DOS) will enter the wait state after a machine check interrupt and logout. The logged data can be printed on the console (using display mode) or printed out with the stand-alone program SEREP. The operator can re-IPL and attempt to continue operations or run microdiagnostics and ASCP prior to requesting CE aid.

- Any control program that contains a recovery routine will enter the routine and attempt execution. As stated in Section 10:05, these routines access model-dependent data and will not operate correctly. Results are unpredictable.

Therefore, control programs containing non-Model 135 recovery routines should be run with the machine check control switch set to the stop after log position.

For the following reasons, it is advantageous for Model 135 users to install an operating system that includes recovery management support designed specifically for the Model 135:

- The number of re-IPL's necessary because of machine malfunctions can be reduced. Most hardware errors can be corrected either by Model 135 hardware recovery procedures or RMS routines. The latter ensure the continuation of system operation whenever possible if the error cannot be corrected. This is particularly important during online operations. In systems without Model 135 RMS routines, instruction retry and single-bit processor and control storage error correction occur, but there is no logout history of these intermittent errors. Other errors will necessitate a re-IPL.

- Status information recorded by RMS routines will assist the customer engineer in the diagnosis of machine malfunctions. This data will be of greatest benefit in diagnosing intermittent errors.

- DOS users who install a DOS supervisor that includes Model 135 RMS will also have the advantages of integrated emulation. (CS/30 cannot be executed on a Model 135.)

- OS MFT control programs that include Model 135 RMS are the only ones that include support of new Model 135 CPU features, block multiplexer channels, the 3330 facility, and integrated emulators.

SECTION 70: COMPARISON TABLES OF HARDWARE FEATURES AND PROGRAMMING SUPPORT - SYSTEM/360 MODELS 25 AND 30 AND SYSTEM/370 MODEL 135

These tables have been included for quick reference. The first compares the hardware features of Models 25, 30, and 135. The second indicates DOS and OS MFT support of Model 135 features.

70:05 HARDWARE FEATURES - SYSTEM/360 MODELS 25 AND 30 AND SYSTEM/370 MODEL 135

Hardware Feature	System/360 Model 25	System/360 Model 30	System/370 Model 135
I. CPU			
A. Internal performance			
1. Relative to Model 30	-	1	2 to 4.5 for commercial 3.5 to 7 for scientific
2. Relative to Model 25	1	-	3.5 to 6.5 for commercial 5.5 to 16 for scientific
B. Instruction set	Standard	Standard	Standard
1. Standard set (Binary arithmetic)			
2. Decimal arithmetic	Standard	Optional	Standard
3. Floating-point arithmetic	Optional	Optional	Optional
4. Extended precision floating point	Not available	Not available	Optional
5. Six new general purpose instructions (MOVE LONG, COMPARE LONG, etc.	Not available	Not available	Standard
6. STORE CPU ID, STORE CHANNEL ID, etc., privileged instructions	Not available	Not available	Standard
C. Interval timer	Optional (16.6-ms resolution)	Standard (16.6-ms resolution)	Standard (3.3-ms resolution)
D. Time of day clock	Not available	Not available	Standard
E. Instruction retry by hardware	No	Limited	Standard
F. Machine check interrupt	Occurs on CPU, main storage, and certain channel errors. One mask bit controls this interrupt.	Same as the Model 25	Occurs after corrected and uncorrected errors. There are five types of machine check and three mask bits.
G. Fixed lower storage area size (including logout area for machine and channel errors)	134 bytes	139 bytes	512 bytes
H. Compatibility features (Optional unless otherwise indicated)	1. 1401/1440/1460 2. 1401/1440/1460 DOS Compatibility (for use with CS/30) 3. Model 20 Mode	1. 1401/1440/1460 2. 1620 (mutually exclusive features)	1. 1401/1440/1460 2. OS/DOS Compatibility (standard)
I. CPU cycle time	900 nanoseconds 1-byte data flow	750 nanoseconds 1-byte data flow	Variable from 275 to 1430 nanoseconds, 2-byte parallel data flow
J. Direct Control feature or External Interrupt feature	Optional (mutually exclusive features)	Optional (mutually exclusive features)	Direct Control is optional (Includes External Interrupt which is not available separately)

Hardware Feature	System/360 Model 25	System/360 Model 30	System/370 Model 135
II. STORAGE			
A. Processor (main) storage sizes	16K 24K 32K 48K	16K 24K 32K - 64K	- - - - - 96K 144K 192K 240K
B. Processor storage cycle	1.8 microseconds for 2 bytes	1.5 microseconds for 1 byte	For CPU operations, 770 nanoseconds read and 935 nanoseconds write for 2 data bytes (includes fetch of next microinsturction). For cycle steal channel operations 660 nanoseconds for 1 or 2 bytes.
C. Processor storage validity checking	Parity checking on each byte. Errors are not corrected by hardware.	Same as Model 25	ECC checking on a doubleword. Single-bit errors are corrected by hardware.
D. Control storage	Reloadable core storage (16K)	Card Capacitor ROS	Reloadable monolithic storage (24K, 36K, or 48K) with ECC
E. Byte boundary alignment (nonprivileged operands)	No	No	Standard
F. Storage and fetch protection	Storage protect is optional. Fetch protect is not available.	Same as Model 25	Standard
III. CHANNELS AND INTEGRATED I/O ATTACHMENTS			
A. Byte multiplexer channel - up to 8 control units	Optional, in addition to integrated attachments. (Cannot be installed if selector channel is present.)	Standard	Standard
1. Subchannels provided	32 for all main storage sizes	96 for all main storage sizes 16K to 64K. (A special feature permits systems with 32K or 64K to have 224 subchannels.)	A total of 16 or 64 subchannels is permitted with any storage size.
B. Selector channels	Optional Maximum of 1, in addition to all integrated I/O attachments. (Cannot be installed with byte multiplexer channel.)	Optional 1 or 2	Optional 1 or 2
1. Maximum individual channel data rate	60 KB	312 KB	1.3 MB for first channel 1.2 MB for second channel
2. Block multiplexer mode	Not available	Not available	Optional for all installed selectors

Hardware Feature	System/360 Model 25	System/360 Model 30	System/370 Model 135
C. Integrated attachments for direct connection of I/O devices to the system without a channel and control unit (does not include integrated attachments for consoles)	A maximum of one each of the following can be attached in addition to a byte multiplexer or a selector channel: 1. 1403 attachment for one 1403 Printer Model 2, 7, or N1 2. 2540 attachment for one 2540 Model 1 3. 2311 attachment for handling up to 4 2311 Model 1 disk storage drives 4. 2560 attachment for one 2560 Multi-Function Machine (used with Model 20 emulation only) 5. Integrated Communications Attachment (ICA) for up to 24 low-speed and 2 medium-speed communications lines. Start/stop and binary synchronous terminals supported by the 2701 control unit can be attached via the ICA.	None available	1. One Integrated File Adapter can be attached to handle from 3 to 5 2314A-type drives (2319 plus 2312 or 2318) 2. One Integrated Communications Adapter can be installed to handle 1 to 8 low- and medium-speed communication lines. Any mixture of provided terminal adapter types (one per line) is permitted. Terminals supported include the 1050, 2740, 2741, 2760, System/7, 2260, 2265 and all binary synchronous terminals supported by the 2701.
D. Channel retry data in a limited channel logout area (ECSW) after channel error, and I/O extended logout data	No	No	Yes
E. Channel-to-Channel Adapter	Not available	Optional	Not available (Model 135 channel can be connected to an adapter installed on a channel on another System/360 or System/370
IV. OPERATOR CONSOLE DEVICES	1. 1052-7 Printer-Keyboard--15 cps (no alter/display mode)	1. Same as Model 25	1. 3210 Model 1 console with alter/display mode (15 cps) 2. 3215 console with alter/display mode (85 cps). (Either a 3210 or 3215 is required.) 3. Alternate and/or additional consoles, such as graphic units, are optional.

Hardware Feature	System/360 Model 25	System/360 Model 30	System/370 Model 135
V. I/O DEVICES			
A. 3211 Printer with tapeless carriage, UCS, and 18 additional print positions	No	Yes	Yes
B. 3803/3420 Magnetic Tape Subsystem	No	Yes except Model 7. Models 3 and 5 cannot be attached to a byte multiplexer channel	Yes (Should not be attached to the byte multiplexer channel.)
C. Other tape units currently announced	All except 2420 models and 2401 Models 3, 5, and 6. (Tape cannot be placed on the byte multiplexer channel.)	All except 2420 Model 7	All can be attached. (May overrun on byte multiplexer channel.)
D. Direct access devices (2311,2314,2319,2303 2301, and 2321)	2311 only (2314 can be attached) via RPQ.)	All except 2303 and 2301 drums. Only channel 1 can have 2314-type facilities attached.	All except 2301 drum. 2314-type disk storage can be attached to the IFA and one channel or to both channels when IFA not present.
E. 3330 facility	No	No	Yes
F. 3505 Card Reader and 3525 Card Punch	No	No	Yes
G. 2560 Multi-Function Card Machine	2560 for Model 20 emulation only	No	No
H. Other devices: 1231, 1285, 1404, 1412, 1418, 1428, 1445, 1827, 2301, 2302, 2305, 2319A2, 3210-2, 7340, 7772, 1052-7, 2150	Yes except 2301, 2302, 2305, 2319A2, 3210-2, 7340, 2150	Yes except 2301, 2305, 2319A2, 3210-2, 7340, 2150	No

70:10 DOS AND OS MFT SUPPORT OF MODEL 135 FEATURES

Hardware Feature	DOS - Model 135	OS MFT - Model 135
I. CPU		
A. Instruction set		
1. Standard set (Binary arithmetic)	All languages	All languages
2. Decimal arithmetic	All languages except FORTRAN	All languages except FORTRAN
3. Floating-point arithmetic	All languages except RPG	All languages except RPG
4. Extended precision floating point	Mnemonics in Assembler D (14K)	Assemblers F and H, and FORTRAN H, PL/I Optimizing Compiler and PL/I Checkout Compiler program products
5. Six new instructions (MOVE LONG, COMPARE LONG, etc.)	Mnemonics supplied for user use in Assembler D (14K).	Mnemonics supplied for user use in Assemblers F and H
6. STORE CPU ID, STORE CHANNEL ID, etc., privileged instructions	Used by the control program	Used by the control program
B. Interval timer	Supported for time intervals and time of day	Supported except for time of day requests
C. Time of day clock	Not supported	Supported for time of day requests
D. Instruction retry by hardware	Both successful and unsuccessful retries logged by MCAR	Both successful and unsuccessful retries logged by MCH
E. Machine check interrupt	Soft and hard machine checks are logged. Recovery procedures are performed.	Same as DOS with some differences in recovery procedures performed.
F. Compatibility features	1. 1401/1440/1460 integrated emulator	1. 1401/1440/1460 integrated emulator 2. OS DOS emulator
G. Direct Control feature	Not supported	Supported by Real Time Monitor
II. STORAGE		
A. Processor (main) storage sizes	All are supported	All are supported
B. Processor storage validity checking (ECC)	When EFL is reached for corrected intermittent single-bit errors, an entry is logged by MCAR. Uncorrected storage errors are also logged and dynamic partition redefinition for processing partitions is attempted so that processing can continue.	MCH logs error as does DOS. MCH attempts to terminate affected task so that processing can continue.
C. Byte boundary alignment (nonprivileged operands)	Programmers can use the byte alignment hardware facility in Assembler programs.	Same as DOS
D. Storage and fetch protection	Storage protect is supported only.	Storage protect is supported only.
III. CHANNELS AND INTEGRATED I/O ATTACHMENTS		
A. Byte multiplexer channel with up to 8 control units and 64 subchannels	Supported	Supported

Hardware Feature	DOS - Model 135	OS MFT - Model 135
B. Selector channels	Supported	Supported
1. Block multiplexer mode	Not supported	Supported
C. Integrated attachments (for direct connection of I/O devices to the system without a channel or control unit)		
1. Integrated File Adapter	1. Supported for same functions as channel attached 2314 storage	1. Supported as DOS
2. Integrated Communications Adapter	2. Supported by BTAM and QTAM for same functions provided for comparable terminal adapters for 2701.	2. Same as DOS
D. Channel retry data in a limited channel logout area (ECSW) after channel error, and I/O extended logout data	CCH routine passes ECSW data to ERP to perform a retry of failing I/O operation if possible. I/O extended logout data is recorded.	Same as DOS
IV. OPERATOR CONSOLE DEVICES		
A. 3210 Model 1 console	Supported as the operating system console	Supported as the primary console
B. 3215 console	Supported as the operating system console	Supported as the primary console
C. Additional consoles, such as graphic units, or specification of an alternate console	Not supported	Additional consoles are supported by MCS and DIDOCS options. An alternate to the primary console can also be specified.
V. I/O DEVICES		
A. 3211 Printer with tapeless carriage, UCS, and 18 additional print positions	Supported	Supported
B. 3803/3420 Magnetic Tape Subsystem	Supported	Supported
C. Other tape units currently announced	Supported	Supported
D. Direct access devices (2311,2314,2319,2303, and 2321)	2303 drums are not supported.	All are supported.
E. 3330 facility with RPS	The 3330 is supported as an I/O device without the RPS capability.	Supported
F. 3505 Card Reader	Supported	Supported
G. 3525 Card Punch	Supported	Supported

READER'S COMMENT FORM

A Guide to the IBM S/370 Model 135 GC20-1738-1

Please comment on the usefulness and readability of this publication, suggest additions and deletions, and list specific errors and omissions (give page numbers). All comments and suggestions become the property of IBM. If you wish a reply, be sure to include your name and address.

COMMENTS

fold fold

fold fold

- Thank you for your cooperation. No postage necessary if mailed in the U.S.A. FOLD ON TWO LINES, STAPLE AND MAIL.

YOUR COMMENTS PLEASE...

Your comments on the other side of this form will help us improve future editions of this publication. Each reply will be carefully reviewed by the persons responsible for writing and publishing this material.

Please note that requests for copies of publications and for assistance in utilizing your IBM system should be directed to your IBM representative or the IBM branch office serving your locality.

fold fold

FIRST CLASS
PERMIT NO. 1359
WHITE PLAINS, N. Y.

BUSINESS REPLY MAIL
NO POSTAGE NECESSARY IF MAILED IN THE UNITED STATES

POSTAGE WILL BE PAID BY...

IBM Corporation
1133 Westchester Avenue
White Plains, N.Y. 10604

Attention: Technical Publications

fold fold

A Guide to the IBM S/370 Model 135 Printed in U.S.A. GC20-1738-1

IBM

International Business Machines Corporation
Data Processing Division
1133 Westchester Avenue, White Plains, New York 10604
[U.S.A. only]

IBM World Trade Corporation
821 United Nations Plaza, New York, New York 10017
[International]

A Guide to the IBM S/370 Model 135 Printed in U.S.A. GC20-1738-1

International Business Machines Corporation
Data Processing Division
1133 Westchester Avenue, White Plains, New York 10604
(U.S.A. only)

IBM World Trade Corporation
821 United Nations Plaza, New York, New York 10017
(International)