1103 CENTRAL EXCHANGE

NEWSLETTER NUMBER 10

December 1956

PX 71900-10

Remington Rand Univac

DIVISION OF SPERRY RAND CORPORATION

1902 WEST MINNEHAHA AVE. ST. PAUL W4, MINNESOTA

PX 71900-10

Title: 1103 CENTRAL EXCHANGE NEWSLETTER NUMBER 10, December 1956

Classification U Publication Date ________ Charge 93888-4

Text pages 0 Photographs ________ Drawings ________

Page No. (of)	Contents	Print Number	Drawing Number	Number
	Title Page	71900-10		
i thru iii	Front Matter (Newsletter)	71900-10		
10-1 thru 10-5	General Card Read-In Routine	71900-10-168	(RW-168)	
10-6 thru 10-15	Eigenvalues of a Tri-Diagonal Matrix	71900-10-169	(RW-169)	
10-16 thru 10-17	Floating Point Card Dump	71900-10-170	(RW-170)	
10-18 thru 10-24	1103 to 1103A Conversion Routine	71900-10-171	(RR-171)	
10-25 thru 10-48	Floating Vector Arithmetic Package	71900-10-172	(RR-172)	
10-49 thru 10-73	Significance Preserving Floating Binary Point Arithmetic for Digital Computers	71900-10-173	(RR-173)	
10-74 thru 10-135	A Method for Generating Random Numbers on the ERA 1103	71900-10-174	(WS-174)	
10-136 thru 10-141	Evaluation of $\|A-\lambda I\|$ for Matrix A (Complex Single Precision Floating Point)	71900-10-175	(RW-175)	
10-142 thru 10-149	Floating Point Linear Matrix Equation Solver (AX=B)	71900-10-176	(RW-176)	
10-150 thru 10-157	Complex Single Precision Floating Point Linear Matrix Equation Solver (AX = B)	71900-10-177	(RW-177)	
10-158 thru 10-169	Complex Gill Method Routine	71900-10-178	(RW-178)	
10-170 thru 10-180	Algebraic Equation Solver	71900-10-179	(RW-179)	
10-181 thru 10-190	Unpacked Floating Point Card Output	71900-10-180	(CV-180)	
10-191 thru 10-210	Continuous Matrix Multiplier Using FLIP III	71900-10-181	(CV-181)	
10-211 thru 10-240	Continuous Matrix Multiplier Using Single or Multi-Precision Arithmetic	71900-10-182	(CV-182)	
10-241 thru 10-330	SPUR - Single Precision Unpacked Rounded Floating Point Package for ERA-1103 Computers	71900-10-183	(CV-183)	

PX 29400

CONTENTS OF PUBLICATION

Title: 1103 CENTRAL EXCHANGE NEWSLETTER 10, December 1956

Classification U Publication Date ______ Charge 93888-4

Text pages 0 Photographs ______ Drawings ______

PX 29400

Newsletter Number 10

December 1956

EDITOR'S PAGE

Correction: On page 9-392 of Newsletter 9, line 2, paragraph 4, of RR-162 should read ". . .will be $p-1 = 2^{35}-32=34,359,738,336$. . .".

Correction: The octal equivalent given for the constant A_4 in the descriptions for the arsin-arcos floating point (RR-75) and the arcsin-arcos stated point (RR-25) routines is in error and should read

37 50417 41233

instead of the listed value of

37 04174 41233.

This correction should be entered in Newsletter 3 (pg. 3-108) for RR-25 and in Newsletter 6 (pg. 6-73) for RR-75.

Editor,
Central Exchange

PX 71900-10-(i)

ENCLOSURES

RW-168	General Card Read-In Routine
RW-169	Eigenvalues of a Tri-Diagonal Matrix
RW-170	Floating Point Card Dump
RR-171	1103 to 1103A Conversion Routine
RR-172	Floating Vector Arithmetic Package
RR-173	Significance Preserving Floating Binary Point Arithmetic for Digital Computers
WS-174	A Method for Generating Random Numbers on the ERA 1103
RW-175	Evaluation of $\|A-\lambda I\|$ for Matrix A(Complex Single Precision Floating Point)
RW-176	Floating Point Linear Matrix Equation Solver (AX=B)
RW-177	Complex Single Precision Floating Point Linear Matrix Equation Solver (AX=B)
RW-178	Complex Gill Method Routine
RW-179	Algebraic Equation Solver
CV-180	Unpacked Floating Point Card Output
CV-181	Continuous Matrix Multiplier Using FLIP III
CV-182	Continuous Matrix Multiplier Using Single or Multi-Precision Arithmetic
CV-183	SPUR - Single Precision Unpacked Rounded Floating Point Package for ERA-1103 Computers
10:27	Computing Center Organization - The Ramo-Wooldridge Corporation
10:28	Mathematical Service Branch - Eglin Field

REVISIONS

CV-154	Unpacked Floating Point Card Read
RW-71	Utility Routine Library
	1. Table of Contents

PX 71900-10-(ii)

PX 71900-10-(iii)

THE RAMO-WOOLDRIDGE CORPORATION
Los Angeles 45, California

General Card Read-In Routine

Specifications

Identification Tag:	CRI-3
Type:	Service Routine (with subroutine entrance)
Special Storage:	The constant pool and temporary pool are not used by this routine
Program Entrance:	40017b
Program Exit:	40020b
Alarm Exit:	The alarm routine is used by this routine

Coded by:	M. Perry	November, 1956
Approved by:	W. F. Bauer	November, 1956

PX 71900-10-(168)

Description

This routine reads cards produced by MDP-1 (binary cards), CPO-0 (fixed point output), CPO-1 (floating point output) and cards key-punched on the 4 field format (described below) for input in floating point, double precision floating point, fixed point, or octal. These input forms may be intermixed on a card, and the cards may be in any sequence desired. The routine automatically differentiates the 2 card forms, and for 4 field cards, recognizes the type of input in each field. All input is rounded properly. This routine reads cards at full card reader speed and loads the memory as directed by the address or addresses appearing on the cards. Once activated, it continues to read cards until it has read and stored a card containing a stop code as described below. The input need not be normalized to retain full significance. The routine stores high speed memory on the drum, operates in high speed memory, and restores high speed memory from the drum prior to leaving the routine.

Operating Instructions

1. When routine is used as a service routine, set PAK to 40017b and start. Routine will read cards until a stop code is recognized, at which time the machine will stop (MSO) with PAK set to 40017b.

2. When routine is used as a subroutine, enter the routine with 37 40020 40017b. Routine will read cards until a stop code is recognized, at which time control is transferred to cell 40020b and hence to the cell following the return jump mentioned above.

   To restore high speed memory at any time, start at 40040b. A and Q will not be restored.

Card Positioning

Card positioning is required before the initial read only. Card reading automatically positions the next card to be read, and a card will be positioned for punching before leaving the routine. If the routine is used as a service routine, the initial positioning must be manual. If the routine is used as a subroutine, initial card positioning should be programmed. This can only be done by one instruction, [EF 00000 vvvvv] where vvvvv contains 00 00000 00114b.

Stop Codes

1. Read stop, 12 col 80. When this code is detected on either a binary card or a four field input card, the routine will stop reading, position a card on the punch side of the bull, and then exit properly (see operating instructions).

2. Machine stop, 12 col 79. When this code is detected on a 4 field input card, the routine will stop reading, position a card on the punch side of the bull, and stop MSO at 72431b. If the machine is started, the exit from the routine will proceed as described in operating instructions.

Either of the above codes may be entered on a blank card and the routine will sense them. The programmer is cautioned not to place the machine stop code on a binary card as this will result in improper loading (see MDP-1 write-up).

Alarm Conditions

1. Binary Cards - As a binary card is read, the words are summed and the result is compared with a sum punched in the card (see MDP-1 write-up). If the sums do not agree, the flexowriter will print

"ALARM 00271 000000000000 0000000vvvvv QQQQQQQQQQQQ"

and the machine will stop. vvvvv is the storage address appearing on the card. The contents of Q are not important. Starting the computer will bypass the alarm, the words will be stored as read, and in the absence of a stop code, reading will proceed.

2. 4 Field Cards - An alarm on a four field card indicates that a number was too large to be entered appropriately into the computer. For floating point numbers (single and double precision) this is equivalent to exponent overflow. For fixed point numbers, this indicates that the input properly scaled and rounded was too large to be entered into a single cell. If an alarm condition is detected, the flexowriter will print

"ALARM 00164 000000000000 0000000vvvvv 000000000000"

and the machine will stop. vvvvv is the storage address of the right-most incorrect number. Any or all of the other numbers may have been incorrect. vvvvv equal to 20,000 indicates that the indicated address field was blank. Starting the computer will cause the correct numbers to be stored, the incorrect numbers to be ignored, and in the absence of a stop code, reading will proceed.

4 Field Card Format

The format described below is one of the formats used by this routine and is the format used by CPO-0, CPO-1, CRI-2, and SNAP Read.

The card columns are designated as follows:

Col 1-4	Identification	This field is ignored by the reading routines
Col 5-23	Field 1	
Col 24-42	Field 2	
Col 43-61	Field 3	
Col 62-80	Field 4	

PX 71900-10-(168)

Each field (except identification) is divided as follows.

digit 1-5	Location or address	(x digit 1 for octal location)
digit 6-15	Value or Mantissa	(x digit 15 for negative value)
digit 16-17	Decimal exponent	(x digit 17 for negative exponent)
digit 18-19	Binary exponent	(x digit 19 for negative exponent)

Addressing Options

The following Addressing options are allowable on the 4 Field input card. Option 1 is used by CPO-0, CPO-1, CRI-2, and SNAP Input.

1. Decimal - Straight conversion to the octal equivalent of the decimal address in the card. No indication is necessary.

2. RAWOOP Decimal - Straight conversion to the octal equivalent of the decimal address except that 40,000 decimal is designated as the first drum address (40,000b). No indication is necessary.

3. Octal - The octal number appearing on the card is the actual address. This must be indicated by an x(11 punch) over the first digit of the address.

4. Blank - A completely blank address field indicates that the number in that field is not to be stored. This is differentiated from an address containing 1 or more zeros which will load cell zero.

Input Numbers

The following varieties of input may be punched on the 4 field input card. They may be in any combination on a card with the exception that for a double precision floating point number the two fields must be consecutive and on the same card. For all input, the decimal point is presumed to be at the extreme left end of the value field. It is recommended that the codes listed below be used for each field. However, cards from CPO-0 and CPO-1 are differentiated by the fact that their "B" (Binary Exponent) fields are non-blank and blank respectively.

1. Floating Point - RS. An input number is designated as floating point by the code RS (Read Snap) in the "B" field. There is no restriction on the "D" field. The resulting floating binary number is rounded and is in the form used by SNAP and by the internal floating point on the 1103A computer. A floating point number consists of 3 parts; a sign bit, followed by an 8 binary bit characteristic biased by 200b, and a 27 bit normalized mantissa. To negate a floating point number, the complete cell is complemented.

2. Double Precision Floating Point - RT. An input number is designated as double precision floating point by the code RT (Read Two) in the "B" field. The "value" portions of 2 consecutive fields are joined to allow 20 decimal digits of input. Both fields must contain addresses but the control information (RT code, algebraic sign, and decimal exponent) is taken from the first field only and ignored on the second field. The resulting floating binary number is rounded and is in the form used by double precision SNAP. The upper cell is identical to a single precision floating point number and the lower cell

PX 71900-10-(168)

consists of a sign bit and 35 binary digits which are an extension of the 27 binary digit mantissa in the upper cell.

The programmer is again cautioned that the two fields concerned must be consecutive and on the same card.

3. Fixed Point - No code. An input number is designated as fixed point by the fact that the "B" field is not blank and does not contain an R. Since the "B" field is used to express the binary scale factor desired, the only caution is that it must not be left blank. 00 (zero zero) must be punched if a scale factor of zero is desired. The binary scale factor is allowed to be negative and the only restriction is that the combination of the B and D fields result in a number which is not too large for a single cell. Normalization is not important since the conversion is done in double precision. The resulting binary number is rounded properly.

4. Octal - RU. An input number is designated as being octal by the code RU (Read Unconverted) in the "B" field. Since 12 octal digits are desirable, the "value" and "D" fields are joined giving 12 digits. Each digit is loaded modulo 8, such that an 8 becomes a zero, and a 9 becomes a 1.

Examples. The following list of input would be punched on two cards. The second card would have a "Stop Read" code in col 80 because of the "+" sign on the last line. The octal address and translation are listed in the comments.

QUANTITY	LOCATION	VALUE	±	D	±	B	±	COMMENTS
	$\bar{4}0122$	120		01		RS		40122 20 14631 46315
	16384	45000000004		25		RU		40000 45 00000 00425
	40064	31415926535	-	01		RT		40100 57 51557 00452
	40065	5897932385						40101 75 67513 47562
	$\bar{0}0123$	0015		04		00		00123 00 00000 00017
	99	15		02		15		00143 00 00017 00000
	100	1	-	01		12		00144 77 77777 67777
	$\bar{0}0122$	15	-	01		RS	+	00122 57 61777 77777

PX 71900-10-(168)

THE RAMO-WOOLDRIDGE CORPORATION
Los Angeles 45, California

Eigenvalues of a Tri-Diagonal Matrix

Specifications

Identification Tag:	EGN-1
Type:	Subroutine available on cards for assembly.
Storage:	145 words of storage needed to assemble this routine. 18 + 2n cells of temporary storage immediately following the temporary pool used, but not stored with subroutine. (n = order of matrix). The constant pool and temporary storage pools are used by this routine.
Regional Addresses Used:	OOR, O1M, O2M, O1R, OOK, OOT, FOO, COO
Entrance and Exit:	RJ OOR01 OOR02 No Punching } See Instructions RJ OOR01 OOR03 Cards Punched } See Instructions
Machine Time:	See table in text.
Mode of Operation:	Floating Complex Arithmetic requiring SNIP be activated.

Coded by:	W. Frank	September 1956
Approved by:	W. Bauer	October 1956

Description

This subroutine computes the n eigenvalues of any real or complex tri-diagonal matrix D, having the form:

$$D = \begin{bmatrix} a_1 & b_1 & 0 & . & . & . & 0 \\ 1 & a_2 & b_2 & . & . & . & 0 \\ 0 & 1 & a_3 & b_3 & . & . & 0 \\ . & & . & & & & . \\ . & & & . & & & . \\ . & & & & a_{n-1} & & b_{n-1} \\ 0 & . & . & . & 1 & & a_n \end{bmatrix}$$

Since complex arithmetic is employed, the elements of the matrix must be presented according to the specification for number representation for use with SNIP. The n elements of the main diagonal, a_i, must be followed by the n-1 elements of the upper adjacent diagonal, b_i, in a region whose initial address is specified by a parameter word.

The more general tri-diagonal matrix J, (also called a "Jacobi Matrix") has all its non-zero elements on the main diagonal and on either of the two immediately adjacent diagonals:

$$J = \begin{bmatrix} a_1 & b_1 & 0 & & . & . & . & 0 \\ c_1 & a_2 & b_2 & 0 & . & . & . & 0 \\ 0 & c_2 & a_3 & b_3 & . & . & . & 0 \\ . & & & . & & & & \\ . & & & & . & & & \\ . & & & & & . & & \\ 0 & . & . & . & . & c_{n-1} & a_{n-1} & b_{n-1} \end{bmatrix}$$

PX 71900-10-(169)

This routine can also treat this case if in place of the n-1 elements of the upper adjacent diagonal, the n-1 products, $c_i b_i$, are supplied.

The subroutine occupies 145 cells and uses the constant and temporary storage pools. In addition 18 + 2n cells of temporary storage must be provided immediately following the Ramo-Wooldridge Temporary Pool. Practical limitations are imposed on n by the available 1024 words of ES storage and the use of SNIP, hence n must be less than 157. In all cases the eigenvalues can be found in the last 2n cells of the (18+2n) cells of temporary storage. In addition the eigenvalues λ_i and associated residues in the characteristic polynomial $P(\lambda_i)$ can be punched on cards. $P(\lambda_i)$ appears in fields one and two while λ_i is in fields five and six. The eigenvalues are also identified serially in the identification field. At the option of the programmer the successive approximations to the λ_i obtained during the iterative process can also be punched with their associated residues in the reduced polynomial. In the event that no punching is desired then a third entrance is available.

Programming Instructions

1. Complex mode of SNIP must be activated.
2. Entrance to the subroutine is made as follows:
   a. RJ 00R01 00R03

      XX 00A00 vvvvv

      where

      00R00 is the location of the first word of the subroutine

      00A00 is the location of the real part of the first element of the diagonal

      vvvvv is the order n of the tri-diagonal matrix

PX (1900-10-(169)

XX gives the option selected

XX = 20, only the eigenvalues λ_i and their respective residues in the characteristic equation are punched.

XX = 00, in addition to the above, the successive approximation to the eigenvalues are punched.

In either case the eigenvalues themselves are stored in the machine starting at the location 51 = 63b.

b. Should no punching of cards be desired then one must use the entrance:

```
RJ  00R01  00R02
20  00A00  vvvvv
```

3. Control is returned to the cell following the parameter word.

<u>Machine Time</u>

The time taken to find the n eigenvalues can be estimated from the following table of empirical times in seconds.

Order of Matrix	No Punching	Punching Eigenvalues	Punching Eigenvalues and Iterations
3	5 Sec.	8 Sec.	13 Sec.
5	17 "	22 "	36 "
8	50 "	58 "	92 "
10	81 "	91 "	143 "
27	776 "	803 "	-
39	2743 "	2782 "	-

PX 71900-10-(169)

Given the tri-diagonal matrix J

$$J = \begin{bmatrix} a_1 & b_1 & 0 & . & . & . & 0 \\ c_1 & a_2 & b_2 & & & & \\ 0 & c_2 & a_3 & b_3 & . & . & \\ & & & . & & & \\ & & & & . & & \\ & & & & & a_{n-1} & b_{n-1} \\ 0 & . & . & . & . & c_{n-1} & a_n \end{bmatrix}$$

The following recursion formula evaluates the characteristic polynomial $P_n(\lambda)$ of J for given λ:

$$P_i(\lambda) = (a_i - \lambda)\, P_{i-1}(\lambda) - b_{i-1}\, c_{i-1}\, P_{i-2}(\lambda) \qquad i = 1, 2, \ldots, n$$

where

$$P_{-1} = 0 \qquad P_o = 1$$

The problem loses no generality by assuming all $c_i = 1$ so that only the a_i and b_i need be given. Alternatively the general problem can be solved by this routine if one supplies $c_i b_i$ in place of b_i.

Using $P_n(-1)$, $P_n(1)$ and $P_n(0)$, the program enters a modified version of the Algebraic Equation Solver (POL-0) and finds, by iteration, the first root λ_1 of $P_n(\lambda)$.

Having found r roots the $(r+1)^{st}$ root is found by considering the polynomial

$$P_{n-r}(\lambda) = \frac{P_n(\lambda)}{\prod_{i=1}^{r} (\lambda - \lambda_i)}$$

where $\lambda_1, \lambda_2, \ldots, \lambda_r$ are the eigenvalues already found. This code prevents the re-computation of a multiple eigenvalue by not allowing λ_{r+1} to approach the value of any of the r roots already found. This, however, did not prohibit the determination of multiple roots in any of the matrices tested, for the reason stated below.

Convergence

A convergence criterion

$$\left| \frac{\lambda_{i+1} - \lambda_i}{\lambda_{i+1}} \right| < 10^{-k}$$

is applied to determine the end of the iteration. In this code k = 6. This gives an accuracy of 6 to 7 places in many low order cases. For large order ($n > 20$) accuracy is reduced since not enough figures are carried in the 27 bit word of SNIP to accurately define the roots after a large number of arithmetic operations have been performed. In the case eigenvalues are repeated the accuracy deteriorates. Furthermore, the residues of the characteristic polynomials in the neighborhood of such a root are exceedingly small. Hence, a second convergence criterion was introduced in order to avoid exponent overflow. If the residue

$$P_{n-r}(\lambda) \leq 2^{-100}$$

then λ_i is accepted as a root. If λ_p has multiplicity p then the code will find p estimates of λ_p such that

a) no two estimates are equal

b) all estimates have residues $\leq 2^{-100}$

If λ_k is an eigenvalue then $\det(D - \lambda_k I)$ should be zero. Inspecting successive iterations and associated residues can therefore give some indication of the convergence of the procedure. The programmer is, however, cautioned in regards

PX 71900-10-(169)

to using this quantity as a measure of accuracy of the root. It is possible, for example, to have a root accurate to 6 places and yet obtain a residue of high order.

D		00R00	00100		00144	00	00000	00000
D		01M00	00118		00166	00	00000	00000
D		02M00	00151		00227	00	00000	00000
D		01R00	00204		00314	00	00000	00000
D		00K00	00239		00357	00	00000	00000
D		00T00	00033		00041	00	00000	00000
D		F0000	00002		00002	00	00000	00000
D		C0000	00003		00003	00	00000	00000
00R00	00	00000	01R16		00144	00	00000	00334
00R01	MJ	00000	00000	EXIT	00145	45	00000	00000
00R02	MJ	00000	02M37	ENTRANCE 1	00146	45	00000	00274
00R03	TP	01R34	02M50	ENTRANCE 2	00147	11	00356	00311
00R04	SP	00R01	00015		00150	31	00145	00017
00R05	TU	A0000	00R07	S	00151	15	20000	00153
00R06	TU	A0000	02M32	E	00152	15	20000	00267
00R07	TP	00000	A0000	T	00153	11	00000	20000
00R08	TP	00013	00T15		00154	11	00015	00060
00R09	TP	00013	00T14	A	00155	11	00015	00057
00R10	TV	A0000	00T15	D	00156	16	20000	00060
00R11	TU	A0000	00R00	O	00157	15	20000	00144
00R12	TP	00K05	Q0000	R	00160	11	00364	10000
00R13	RS	00T15	00016	N-1 E	00161	23	00060	00020
00R14	LA	A0000	00016	2N-2 S	00162	54	20000	00020
00R15	QS	A0000	01R06	S	00163	53	20000	00322
00R16	TP	00T15	00T16	E	00164	11	00060	00061
00R17	TN	00016	00T17	S	00165	13	00020	00062
01M00	RA	00R01	00016		00166	21	00145	00020
01M01	RP	10005	01M03	SET	00167	75	10005	00171
01M02	TP	00013	00T09		00170	11	00015	00052
01M03	TP	00K01	00T12	UP	00171	11	00360	00055
01M04	TN	00K00	00T08		00172	13	00357	00051
01M05	TP	00K00	00T10	STARTING	00173	11	00357	00053
01M06	TN	00K01	00T06		00174	13	00360	00047
01M07	TP	00013	00T07	VALUES	00175	11	00015	00050
01M08	RJ	01R30	01R00		00176	37	00352	00314
01M09	TP	00029	00T00		00177	11	00035	00041
01M10	TP	00030	00T01		00200	11	00036	00042
01M11	TP	00K01	00T06		00201	11	00360	00047
01M12	RJ	01R30	01R00		00202	37	00352	00314
01M13	TP	00029	00T02		00203	11	00035	00043
01M14	TP	00030	00T03		00204	11	00036	00044
01M15	TP	00013	00T06		00205	11	00015	00047
01M16	RJ	01R30	01R00		00206	37	00352	00314
01M17	TP	00029	00T04		00207	11	00035	00045
01M18	TP	00030	00T05		00210	11	00036	00046
01M19	MJ	00000	02M31		00211	45	00000	00266
01M20	LDMP	00T00	00T08		00212	14	30041	14051
01M21	ADST	00T04	00023		00213	14	04045	34027
01M22	MPST	00T08	00025		00214	14	14051	34031
01M23	LDMP	00T02	00T10		00215	14	30043	14053
01M24	STSU	00027	00023	S	00216	14	34033	11027

PX 71900-10-(169)

```
01M25 LDMP 00T04 00T10                              00217  14 30045 14053
01M26 ADNO F0000 00000                              00220  14 04002 00000
01M27   TN F0000 00029                              00221  13 00002 00035
01M28   TN C0000 00030                              00222  13 00003 00036
01M29 ADMP F0000 00T08                              00223  14 04002 14051
01M30 MPLD 00023 00T04    S                         00224  14 15027 30045
01M31 SUMP 00027 00T10                              00225  14 10033 14053
01M32 ADMP 00025 F0000    S                         00226  14 05031 14002
02M00 ADRT 00023 00023       S                      00227  14 04027 51027
02M01   TN C0000 C0000                              00230  13 00003 00003
02M02 LDMP F0000 00025                              00231  14 30002 14031
02M03   SJ 02M04 02M06                              00232  46 00233 00235
02M04   TN 00023 00023          F                   00233  13 00027 00027
02M05   TN 00024 00024           I                  00234  13 00030 00030
02M06 LDAD 00025 00023       S    N                 00235  14 30031 05027
02M07 PMNO 00000 00000            D                 00236  14 24000 00000
02M08   ZJ 02M10 02M09                              00237  47 00241 00240
02M09   TP 00K01 00023                              00240  11 00360 00027
02M10 LDDV 00029 00023           ITERANT            00241  14 30035 20027
02M11 STMP 00T08 00T12       S                      00242  14 34051 15055
02M12 ADNO 00T06 00000    S      AND                00243  14 05047 00000
02M13   RJ 01R30 01R00                              00244  37 00352 00314
02M14 DVPM 00T04 00000           FUNCTIONAL         00245  14 20045 24000
02M15   TJ 00K02 02M20                              00246  42 00361 00253
02M16   TN 00K00 F0000           VALUE              00247  13 00357 00002
02M17   TP 00013 C0000                              00250  11 00015 00003
02M18 LDMP F0000 00T08                              00251  14 30002 14051
02M19   MJ 00000 02M11                              00252  45 00000 00242
02M20   TP 00K01 F0000                              00253  11 00360 00002
02M21 ADNO 00T08 00000                              00254  14 04051 00000
02M22   TP F0000 00T10                              00255  11 00002 00053
02M23   TP 00T09 00T11                              00256  11 00052 00054
02M24   RP 30004 02M26          SET UP              00257  75 30004 00261
02M25   TP 00T02 00T00                              00260  11 00043 00041
02M26   TP 00029 00T04            FORNEXT           00261  11 00035 00045
02M27   TP 00030 00T05                              00262  11 00036 00046
02M28 LDDV 00T12 00T06            ITERATION         00263  14 30055 20047
02M29 PMNO 00000 00000                              00264  14 24000 00000
02M30   TJ 00K03 02M40          CONVERGED           00265  42 00362 00277
02M31   TP 02M29 A0000                              00266  11 00264 20000
02M32   TJ 00000 01M20                              00267  42 00000 00212
02M33   RP 30004 02M35                              00270  75 30004 00272
02M34   TP 00T04 00006                              00271  11 00045 00006
02M35 PDNO 00010 00000                              00272  14 74012 00000
02M36   MJ 00000 01M20                              00273  45 00000 00212
02M37   TP 02M39 02M50                              00274  11 00276 00311
02M38   MJ 00000 00R04                              00275  45 00000 00150
02M39   MJ 00000 02M51                              00276  45 00000 00312
02M40   TP 00T14 00004                              00277  11 00057 00004
02M41 LDST 00T06 00008                              00300  14 30047 34010
02M42 STST 00025 00T18      B                       00301  14 34031 36063
```

```
02M43   RA  00T14 00K04                        00302 21 00057 00363
02M44   RJ  01R15 01R01                        00303 37 00333 00315
02M45   TP  F0000 00006                        00304 11 00002 00006
02M46   TP  C0000 00007                        00305 11 00003 00007
02M47   RA  00T17 00016                        00306 21 00062 00020
02M48   SA  00016 00015                        00307 32 00020 00017
02M49   TU  A0000 00005          PUNCH         00310 15 20000 00005
02M50 PDPD  00010 00010          ITERANT       00311 14 74012 74012
02M51   IJ  00T16 01M01                        00312 41 00061 00167
02M52   MJ  00000 00R01                        00313 45 00000 00145
01R00   TV  00R00 01R15                        00314 16 00144 00333
01R01   RP  10003 01R03                        00315 75 10003 00317
01R02   TP  00013 00030          E             00316 11 00015 00036
01R03   TP  00K01 00029           V            00317 11 00360 00035
01R04   TU  00R00 00004                        00320 15 00144 00004
01R05   TP  00T15 00023             L          00321 11 00060 00027
01R06 LDMP  00000 00031  B    S      U         00322 14 32000 15037
01R07 LDSU  00000 00T06  B                     00323 14 32000 10047
01R08 MPSU  00029 00031                T       00324 14 14035 10037
01R09   TP  00029 00031                 E      00325 11 00035 00037
01R10   TP  00030 00032                  S     00326 11 00036 00040
01R11   TP  F0000 00029          C             00327 11 00002 00035
01R12   TP  C0000 00030           H            00330 11 00003 00036
01R13   RA  00004 00K04            A           00331 21 00004 00363
01R14   IJ  00023 01R06             R          00332 41 00027 00322
01R15   MJ  00000 00000              A         00333 45 00000 00000
01R16   TP  00T17 A0000               C        00334 11 00062 20000
01R17   SJ  01R28 01R18               T        00335 46 00350 00336
01R18   TP  00T17 00023                 E      00336 11 00062 00027
01R19   TP  00K01 00031                R       00337 11 00360 00037
01R20   TP  00013 00032           ISTIC        00340 11 00015 00040
01R21   TP  00013 00004                        00341 11 00015 00004
01R22 LDSU  00T06 00T18       B                00342 14 30047 12063
01R23 MPPM  00031 00000    S                   00343 14 15037 24000
01R24   ZJ  01R25 01R31           EQUATION     00344 47 00345 00353
01R25   RA  00004 00K04                        00345 21 00004 00363
01R26   IJ  00023 01R22                        00346 41 00027 00342
01R27 LDDV  00029 00031                        00347 14 30035 20037
01R28 STPM  00029 00000                        00350 14 34035 24000
01R29   ZJ  01R30 02M40          CONVERGED     00351 47 00352 00277
01R30   MJ  00000 00000                        00352 45 00000 00000
01R31   TP  00K00 F0000                        00353 11 00357 00002
01R32 ADNO  00T06 00000    S                   00354 14 05047 00000
01R33   MJ  00000 01R01                        00355 45 00000 00315
01R34 PDPD  00010 00010                        00356 14 74012 74012
00K00   05  00000 00000 -01 F    C             00357 20 04000 00000
00K01   01  00000 00000     F     O            00360 20 14000 00000
00K02   01  00000 00000   1 F      N           00361 20 45000 00000
00K03   01  00000 00000 -06 F       S          00362 15 54143 36750
00K04   00  00002 00000              T         00363 00 00002 00000
00K05   00  00777 00000 B             ANTS     00364 00 00777 00000
START                                          00000 45 00000 00000
```

PX 71900-10-(169)

THE RAMO-WOOLDRIDGE CORPORATION
Los Angeles 45, California

Floating Point Card Dump

Specifications

Identification Tag:	MDP-5
Type:	Service Routine (with subroutine entrance)
Service Entrance:	Address 40024 b
Program Entrance and Exit:	37 40020 40024 b
Other Routines Used:	This routine uses MDP-4 and SNAP Output

Coded and Checked by:	R. Beach	October, 1956
Approved:	W. F. Bauer	October, 1956

PX 71900-10-(170)

Description:

This routine operates exactly the same as MDP-4 (Octal Card Dump) with the following exceptions:

1. Entrance Address is 40024 b

2. Output is floating decimal on SNAP output cards. Addresses are five digit octal numbers with leading zeros suppressed.

3. A parameter word of zero will dump cells 00000 - 00777 b.

The routine treats each word to be dumped as a floating point (SNAP form) number and converts it to a floating decimal number. Non SNAP numbers (i.e. instructions and fixed point numbers) may be included among the words to be dumped but their converted values will be meaningless.

The listing will be double-spaced; however, if a card was omitted because it would have contained all zeros, no additional spacing is provided on the listing.

The routine is essentially a driver for the SNAP output routine and MDP-4, modifying each so that MDP-4 uses the SNAP output routine instead of its octal output section.

PX 71900-10-(170)

REMINGTON RAND UNIVAC

ST. PAUL DEPARTMENT--INFORMATION SCIENCE

18 December 1956

1103 TO 1103A CONVERSION ROUTINE

I. TYPE: Service routine or subroutine.

II. STATUS: Code checked and machine checked by Bill Wallace.

III. PURPOSE: This routine changes A and Q machine addresses from 20000 and 10000 to 32000 and 31000 respectively, and detects magnetic tape and external function instructions. Various options are provided for print out of those addresses where an A or Q reference is modified, (indicating also u or v portion) and punching the converted program in bioctal or flex code.

IV. USAGE:

A. STORAGE REQUIRED: The program is coded in RECO form and it is therefore possible to operate the program from a location providing 320 octal drum address and 2000 additional octal drum addresses for a HSS image region. Such a location of the program and image region is done by assigning the desired starting addresses to regions BB and IR respectively, of the reco tape (see coding) all other regions being in HSS, and hence remaining the same. The regional assignment can be on a separate tape from the main program reco tape, but this tape should have END. c.r. at the end. (See RECO write-up.)

In addition to the RECO tapes, a bioctal tape of the program is available where the program is stored at 66000-66320, with the image region 76000-77777.

B. INPUT-OUTPUT: Output is a punched tape in bioctal or flex code of the changed program if desired. Also the following is printed out as the conversion routine is operating: (This is also optional.) u aaaaa or vaaaaa, where aaaaa is the address where an A or Q reference has been modified and u or v shows whether the u or v address of the instruction has been modified. Also, TAPE is printed out when an 1103 magnetic tape instruction is encountered, and EF and address when an external function command occurs.

C. OPERATING INSTRUCTIONS:

(1) Used as a service routine proceed as follows: (the term "program" here refers to a program to be converted.)

a) Master clear, MD start
b) Set PAK to 66000, (or bb)
c) Insert in Q_u the first address of the program
d) Insert in Q_v the last address of the program
e) Insert in v address of A_R the address of the last instruction of the program, or the last address of the program wherein one wishes to have A and Q addresses modified.

PX 17900-10-(171)

f) Insert in v address of A_L the following codes for the various options:

00000	bioctal punch of converted program and print out of addresses where modification occurs.
00001	same as above but no print out.
00002	flex code punch of converted program and print out.
00003	same as above, but no print out.
00004	print out, but no punch of converted program.
00005	no print out and no punch of converted program.

A 56 0 66010 (bb10) stop occurs if a gross error is made in the set-up, e.g. transposition of Q_u and Q_v.

(2) Used as a subroutine, proceed as follows:

a) Program the transfer of parameter as listed above to the A and Q registers.
b) Execute the instruction:
   RJ bb2 bb
c) The options are selected in the same manner as previously shown.

(3)

a) The use of this conversion routine assumes that the program to be converted is stored either all in core storage or all in drum storage.
b) The conversion routine is coded for operation on either an 1103A, or on the 1103 (Serial 9) at RRU, St. Paul.

BW:pac

PX 71900-10-(171)

V. CODING

A. Regions

```
re bb66000      re ua124
re ir76000      re va137
re ff30000      re up155
re cr0          re pr162
re cb33         re tp173
re cd54         re ef200
re cf71         re cs205
re kk75         re dd212
re mm112        re tt310
```

B. Program

```
bb0        45  0      bb10     Entrance
  1        56  0      bb10     Error stop
  2        45  0      (ff)     Subroutine exit
  3        0   0      0        Storage first address
  4        0   0      0        Storage last address
  5        0   0      0        Storage initial A
  6        0   0      0        Storage initial Q^r
  7        45  0      ff       Constant
 10        tp  Q      bb6
111        lt  10000  bb5
 12        lt  00000  A
 13        tp  0      ir
 14        tp  bb     0
 15        rp  31777  bb17
 16        tp  1      ir1      Store HSS
 17        rp  30400  cb       To start of core program
 20        tp  bb1    cr1      Conclusion of program
 21        rp  31777  bb23
 22        tp  ir1    1        Restore HSS
 23        tp  ir     0
 24        tp  bb6    Q        Restore Q for dump
 25        tp  bb7    A
 26        ej  bb2    bb31     Test, subr. or serv?
 27        rj  70036  (70006)  No, subr.
 30        45  0      bb2      To exit
 31        rj  70036  (70006)
 32        56  00000  bb

 33  cb0   ej  dd43   cd       No punch
 34    1   ej  dd64   cd2      Punch flex
 35    2   ej  dd65   cd5      No print, no punch
 36    3   ej  dd66   cd13     No print, punch flex
 37    4   ej  dd60   cd6      No print, punch biootal
 40    5   tp  cr5    A        Last address
```


PX 71900-10-(171)

41	6	tj	dd	cf	HSS?
42	7	qt	dd1	cr4	Store first address
43	10	lq	q	25	
44	11	qt	dd1	cr3	Store last address
45	12	tv	cr3	mm	Set up transfer
46	13	la	cr3	20017	of Modified Contents
47	14	tu	A	kk	Set up first address
50	15	ra	cr4	dd60	to be modified
51	16	st	cr3	tt	No. of words
52	17	ij	tt	kk	
53	20	45	0	bb1	Error
54	cd0	tv	dd67	mm4	
55	1	45	0	cb5	No punch
56	2	rs	bb27	dd60	
57	3	rs	bb31	dd60	
60	4	45	0	cb5	Flex punch
61	5	rj	cd1	cd	
62	6	tv	up4	ua10	
63	7	tv	up4	ua12	
64	10	tv	va15	va7	
65	11	tp	va15	va11	No print
66	12	45	0	cb5	
67	13	rj	cd4	cd2	
70	14	45	0	cd6	
71	cf0	ra	cr5	dd3	Add 76000 to V
72	1	ra	Q	dd2	Add 76000 to U and V
73	2	tp	cs	pr	Arrange to print core address
74	3	45	0	cb7	
75	kk0	tp	(ff)	Q	
76	1	tp	Q	tt1	
77	2	qt	dd5	tt3	Mask operation code
100	3	tp	tt3	A	
101	4	ej	dd10	ef	External function
102	5	ej	dd11	mm	Final stop
103	6	ej	dd12	mm	Interpret
104	7	rp	20014	kk11	Commands where
105	10	ej	dd13	tp	V only to be modified and tape commands
106	11	rp	20004	kk13	Split instruction,
107	12	ej	dd27	mm10	Modify U only
110	13	rj	ua6	ua	Modify U
111	14	rj	va5	va	Modify V
112	mm0	tp	tt1	(tt1)	Transfer modified Content
113	1	tp	kk	A	
114	2	lt	25	A	Obtain current
115	3	rs	dd6	A	Address
116	4	ej	cr5	bb21	Test, end of
117	5	ra	kk	dd7	Modifiable address

PX 71900-10-(171)

```
120   6     ra  mm     dd60        Add 1
121   7     45  0      eb17
122   10    rj  ua6    ua          Modify U only
123   11    45  0      mm

124   ua0   tp  tt1    Q
125   1     lq  Q      25
126   2     qt  dd33   tt4         Mask 1st octal digit
127   3     tp  tt4    A
130   4     ej  dd34   ua7         Q?
131   5     ej  dd35   ua11        A?
132   6     45  0      ff
133   7     ra  tt1    dd36        Add 21000
134   10    45  0      up          To print
135   11    ra  tt1    dd37        Add 12000
136   12    45  0      up

137   va0   tp  tt1    Q
140   1     qt  dd33   tt4
141   2     tp  tt4    A
142   3     ej  dd34   va6         Q?
143   4     ej  dd35   va10        A?
144   5     45  0      ff
145   6     ra  tt1    dd40        Add 21000
146   7     45  0      va11        To print
147   10    ra  tt1    dd41        Add 12000
150   11    pr  0      dd42        Carriage return
151   12    pr  0      dd43        Space
152   13    pr  0      dd44        "V"
153   14    rj  pr10   pr
154   15    45  0      va5

155   up0   pr  0      dd42        Carriage return
156   1     pr  0      dd45        "U"
157   2     pr  0      dd43        Space
160   3     rj  pr10   pr
161   4     45  0      ua6

162   pr0   tp  kk     Q
163   1     lq  Q      6
164   2     tp  dd43   tt2         Index
165   3     lq  Q      3
166   4     qt  dd46   A
167   5     at  dd47   pr6         Print digit
170   6     (pr 0      ff)
171   7     ij  tt2    pr3
172   10    45  0      ff

173   tp0   rp  20004  kk14        Test for tape
174   1     ej  dd20   tp2           Instructions
```

```
175    2    pr  0      dd42        Carriage return
176    3    rp  10004  mm
177    4    pr  0      dd60        Print "tape"

200  ef0    pr  0      dd42        Carriage return
201    1    rp  10005  ef3
202    2    pr  0      dd70        Print "EF"
203    3    rj  pr10   pr          Print address
204    4    45  0      kk14        To V address modification

205  cs0    rj  cs4    cs1
206    1    tu  kk     tt5
207    2    rs  tt5    dd4         Subtract 76000 from V
210    3    tp  tt5    Q
211    4    45  0      ff

212  dd0    0   0      02000
213    1    0   0      77777
214    2    0   76000  76000
215    3    0   0      76000
216    4    0   76000  0
217    5    77  0      0
220    6    0   11     0
221    7    0   1      0
222   10    ef  0      0
223   11    fs  0      0
224   12    ip  0      0
225   13    lt  0      0
226   14    45  0      0
227   15    56  0      0
230   16    pr  0      0
231   17    pu  0      0
232   20    rm  0      0
233   21    wm  0      0
234   22    am  0      0
235   23    bm  0      0
236   24    rp  0      0
237   25    er  0      0
240   26    ew  0      0
241   27    sp  0      0
242   30    sa  0      0
243   31    sn  0      0
244   32    ss  0      0
245   33    0   0      70000
246   34    0   0      10000
247   35    0   0      20000
250   36    0   21000  0
251   37    0   12000  0
252   40    0   0      21000
```

PX 71900-10-(171)

253	41	0	0	12000	
254	42	0	0	45	Carriage return
255	43	0	0	4	Space
256	44	0	0	17	V
257	45	0	0	34	U
260	46	0	0	7	
261	47	61	0	dd50	
262	50	0	0	37	Flex code 0
263	51	0	0	52	1
264	52	0	0	74	2
265	53	0	0	70	3
266	54	0	0	64	4
267	55	0	0	62	5
270	56	0	0	66	6
271	57	0	0	72	7
272	60	0	0	1	flex code t
273	61	0	0	30	flex code a
274	62	0	0	15	flex code p
275	63	0	0	20	flex code e
276	64	0	0	2	
277	65	0	0	5	
300	66	0	0	3	
301	67	0	0	bb32	
302	70	0	0	47	Shift up
303	71	0	0	20	"E"
304	72	0	0	26	"F"
305	73	0	0	57	Shift down
306	74	0	0	4	
307	75	0	0	0	Not used
310	tt0	0	0	0	Temporary storage (310–315)
311	1	0	0	0	
312	2	0	0	0	
313	3	0	0	0	
314	4	0	0	0	
315	5	0	0	0	

BW:pac

PX 71900-10-(171)

Remington Rand Univac

Floating Vector Arithmetic Package

GENERAL DESCRIPTION

This package contains four subroutines: vector roll-off, vector unpack, scalar product of two vectors, and vector sum. Each of the subroutines is self-contained and can be used independently of the others. These operations are performed on arbitrarily located vectors of not more than 108 elements. The arithmetic is floating point with one biased characteristic serving for all of the elements of a vector. The bias of the characteristic is $40{,}000_{(8)}$.

PACKED FORM OF A VECTOR

Associated with every vector, $\bar{X} = (x_1, \ldots, x_n)$, is a set of numbers, $(b_1, \ldots, b_n)$, each element of which is either 0 or 1. This set of numbers is defined in the following way: if $x_i=0$, then $b_i=0$; if $x_i \neq 0$, then $b_i=1$. The three binary numbers,

$$b_1 x 2^{35} + b_2 x 2^{34} + \ldots + b_{36},$$
$$b_{37} x 2^{35} + b_{38} x 2^{34} + \ldots + b_{72},$$
$$b_{73} x 2^{35} + b_{74} x 2^{34} + \ldots + b_{108},$$

where $b_{n+1}=b_{n+2}=\ldots=b_{108}=0$ if $n \angle 108$, are called the Q-words of the vector $\bar{X}$. It is clear that a vector is well-defined if the Q-words, the number of elements, and an ordered list of the non-zero elements are given.

The operand vectors of the floating vector subroutines must be packed (or stored) in the manner which we now describe. The first three addresses of a vector storage location are occupied by the three Q-words.

PX 71900-10-(172)

The mantissae of the non-zero elements of the vector are stored sequentially in the addresses immediately following the address of the last Q-word. These mantissae are scaled so that the numerically largest has 32 binary digits. There are no blank addresses between successive vectors.

Each vector has a keyword. The v-address of the keyword contains the biased characteristic of the vector; the u-address contains the starting address (address of the first Q-word) of the vector. The address of the keyword of a vector is called the directory address of that vector; the aggregate of all the directory addresses of a system of vectors is called the directory of that system. The keywords are stored in the same order as the corresponding vectors, and there are no blank addresses between successive keywords. Following the last keyword in the directory is a pseudo keyword. If the last non-zero mantissa of the last stored vector is in location y, then y+1 is entered in the v-address of the pseudo-keyword.

NOTATION

1. Throughout the subroutines three blocks of addresses are utilized for vector work spaces and temporary storage. We shall refer to these blocks as R, S, and T. By R_i we will mean the i^{th} address in the block R. We denote by m the number of elements in the operand vectors.

PROGRAM PARAMETERS

1. Locations 00005 through 00017 are reserved for constants and program parameters.

PX 71900-10-(172)

The following parameters must be provided by the programmer:

00005: R_1=starting address of R

00006: S_1=starting address of S

00007: T_1=starting address of T

00010: m=number of elements in operand vectors.

The following parameters are provided by the various subroutines:

00011: Keyword of vector in R

00012: Keyword of next vector in the directory

00013: (u-address) 3+number of non-zero elements of vector in R

00014: Keyword of vector in S

00015: Keyword of vector in T

00016: (u-address) 3+number of non-zero elements of vector in S

00017: (u-address) 3+number of non-zero elements of vector in T.

In the event that the programmer wishes to use some of the routines without using the whole package, it will be necessary for him to provide some of the parameters in this last group.

ROLL-OFF

The roll-off subroutine transfers a packed vector, $\overline{X}$, including Q-words, from permanent storage to R. It is coded in standard form with one exit and one entrance, and is assembly modifiable.

UNPACK

The unpack subroutine unpacks (i.e., provides all zero elements that were omitted in the packed vector) the vector, $\overline{X}$, contained in R and leaves the result in either S or T, depending on which entrance is selected. It is coded in standard form with one exit and two entrances, and is assembly modifiable.

PX 71900-10-(172)

Before entering the unpack subroutine, $\overline{X}$ must be rolled-off in R. The roll-off provides all program parameters.

SCALAR PRODUCT

If $\overline{X}=(x_1,\ldots,x_n)$ and $\overline{Y}=(y_1,\ldots y_n)$, then the scalar product, $\overline{X}\cdot\overline{Y}$, is defined by

$$\overline{X}\cdot\overline{Y}=\sum_{i=1}^{m} x_i y_i.$$

The scalar product subroutine forms the scalar product of the vector in S and the vector in T. It scales the mantissa of this product so that it contains 32 binary digits.

If either $x_i=0$ or $y_i=0$ then, of course, the term $x_i y_i$ contributes nothing to the scalar product; in this case the subroutine avoids formation of the term $x_i y_i$.

We give a brief explanation of how the Scalar product is formed. Associated with the sum $\overline{X}\cdot\overline{Y} = \sum_{i=1}^{m} x_i y_i$ are three words

$$C_1:\ c_1 x 2^{35}+c_2 x 2^{34}+\ldots+c_{36},$$
$$C_2:\ c_{37} x 2^{35}+c_{38} x 2^{34}+\ldots+c_{108},$$

analogous to the Q-words of a vector (i.e., if $x_i y_i=0$, then $c_i=0$, and if $x_i y_i\neq 0$, then $c_i=1$; and if $n<108$, then $c_{n+1}=\ldots=c_{108}=0$). It is clear that if A_1, A_2, A_3 are the Q-words of $\overline{X}$, and if B_1, B_2, B_3 are the Q-words of $\overline{Y}$, then

$$C_i = L(A_i)(B_i),\ i=1,\ 2,\ 3.$$

The subroutine forms C_1, C_2, C_3; it then stores sequentially in S all those x_i, and in T all those y_i, for which $c_i=1$; finally, with a repeated multiply add instruction, it forms the sum.

PX 71900-10-(172)

$$\overline{X}\cdot\overline{Y} = \sum_{i=1}^{k} (S_i)\ (T_i),$$

where k is the number of non-zero terms in the sum

$$\sum_{i=1}^{m} x_i y_i.$$

Before entering this subroutine the vectors $\overline{X}$ and $\overline{Y}$ must be unpacked in S and T respectively. The unpack subroutine provides all program parameters.

VECTOR SUM

The sum of two vectors $\overline{X}=(x_1,\ldots,x_n)$ and $\overline{Y}=(y_1,\ldots,y_n)$ is defined by

$$\overline{X}+\overline{Y} = (x_1+y_1,\ldots,x_n+y_n).$$

The vector sum subroutine forms the sum of the vector in S and the vector in T and leaves the result in S.

The subroutine compares the characteristics of $\overline{X}$ and $\overline{Y}$; it then shifts right the mantissae of the vector with the smaller characteristic a number of bits equal to the absolute value of the difference of the characteristics; next it adds corresponding mantissae of the two vectors; finally it shifts the mantissae of the sum until the largest has 32 binary digits.

Before entering this subroutine, the vectors $\overline{X}$ and $\overline{Y}$ must be unpacked in S and T respectively. The unpack subroutine provides all necessary parameters.

PX 71900-10-(172)

Roll-Off

Author: P. Nikolai Date: 27 August 1956

Type: subroutine

Code Check by: R. C. Gunderson

Machine Check by: P. Nikolai

Correction of Routine: R. C. Gunderson

Revision of Routine: C. D. Dixon

Form: Standard 1103 Subroutine;
Assembly modifiable

Exit: 01001

Entrance: 01002

Storage Required:

Instructions: 01000 through 01020

Constants: 01023 through 01023

Temporary Storage: 01021 through 01022

Number of commands for assembly modification: 240

Preliminary settings: The directory address of the vector to be rolled-off must be placed in the u-address of the accumulator.

Final Results:

(R)=Q-words and non-zero mantissae of vector

(00011)=Keyword of vector

(00012)=Next keyword in directory

(00013)=3+number of non-zero elements of vector

Time: 3.64 milliseconds, maximum

Unpack

Author: P. Nikolai Date: 27 August 1956

Type: subroutine

Code Check by: R. C. Gunderson

Machine Check by: P. Nikolai

Correction of Routine:

Revision of Routine: C. D. Dixon

Form: Standard 1103 Subroutine
Assembly modifiable

Exit: 01001

Entrance for unpack in T: 01002

Entrance for unpack in S: 01003

Storage Required:

Instructions: 01000 through 01046

Constants: 01047 through 01054

Temporary Storage: 01055 through 01056

Number of commands for assembly modification: $47_{(8)}$

Preliminary Settings: Before entering the unpack subroutine, the vector, $\bar{x}$, must be rolled-off in R. The roll-off subroutine provides all program parameters.

Final Results:

for unpack in S,

(S)=Q-words and mantissae of $\bar{x}$

(00014)=Keyword of $\bar{x}$

(00016)=3+number of non-zero elements of $\bar{x}$

PX 71900-10-(172)

for unpack in T,

(T)=Q-words and mantissae of $\bar{x}$

(00015)=Keyword of $\bar{x}$

(00017)=3+number of non-zero elements of $\bar{x}$

Operating instructions:

1. Directory address of $\bar{x} \longrightarrow A_u$.
2. Enter Roll-off subroutine; roll-off $\bar{x}$ in R.
3. Enter unpack subroutine at 01002 (01003); unpack $\bar{x}$ in T(S).

Time: 32.1 milliseconds, maximum

Scalar Product

Author: E. Feller   Date: 27 August 1956

Type: subroutine

Code Check by: R. C. Gunderson

Machine Check by: E. Feller

Correction of Routine: R. C. Gunderson

Revision of Routine: C. D. Dixon

Form: Standard 1103 Subroutine;
Assembly modifiable

Exit: 01001

Entrance: 01002

Storage required:

Instructions: 01000 through 01124

Constants: 01125 through 01141

Temporary Storage: 01142 through 01152

Number of commands for assembly modification: $125_{(8)}$

Preliminary settings: Before entering this subroutine the vectors $\overline{x}$ and $\overline{y}$ must be unpacked in S ant T respectively. The unpack subroutine provides all program parameters.

Final Results:

(A)=(01152)= mantissa of $\overline{x}\cdot\overline{y}$.

(Q)=(01151)= characteristic of $\overline{x}\cdot\overline{y}$.

Accuracy: 32 bits

Other subroutines used: Roll-off
Unpack

PX 71900-10-(172)

Operating Instructions:

1. Directory of $\overline{x} \longrightarrow A_u$.
2. Enter Roll-off subroutine; foll-off $\overline{x}$ in R.
3. Enter Unpack subroutine at 01003; unpack $\overline{x}$ in S.
4. Directory address of $\overline{y} \longrightarrow A_u$.
5. Enter Roll-off subroutine; roll-off $\overline{y}$ in R.
6. Enter Unpack subroutine at 01002, unpack $\overline{y}$ in T.
7. Enter Scalar product subroutine; for $\overline{x} \cdot \overline{y}$.

Time:

Vector Sum

Author: P. Nikolai

Date: 27 August 1956

Type: subroutine

Code check by: R. Gunderson

Machine Check by: P. Nikolai

Correction of Routine: R. Gunderson

Revision of Routine: C. D. Dixon

Form: Standard 1103 Subroutine
Assembly modifiable

Exit: 01001

Entrance: 01002

Storage Required:

Instructions: 01000 through 01132

Constants: 01133 through 01142

Temporary Storage: 01143 through 01146

Number of Commands for assembly modification: $133_{(8)}$

Preliminary settings: Before entering this subroutine, the vectors $\overline{x}$ and $\overline{y}$ must be unpacked in S and T respectively. The unpack subroutine provides all program parameters.

Final Results:

(S)=mantissae of $\overline{x}+\overline{y}$. The routine does not provide Q-words for the vector sum.

(00014)=characteristic of $\overline{x}+\overline{y}$.

Accuracy: maximum 32 bits

Other Subroutines used: Roll-off
Unpack

PX 71900-10-(172)

Operating Instructions:

1. Directory address of $\overline{x} \longrightarrow A_u$.
2. Enter Roll-off subroutine; roll-off $\overline{x}$ in $\bar{R}$.
3. Enter Unpack subroutine at 01003; unpack $\overline{x}$ in S.
4. Directory address of $\overline{y} \longrightarrow A_u$.
5. Enter Roll-off subroutine; roll-off $\overline{y}$ in R.
6. Enter unpack subroutine at 01002; unpack $\overline{y}$ in T.
7. Enter Vector Sum subroutine; form $\overline{x}+\overline{y}$.

Time: 37.5 milliseconds, maximum

PX 71900-10-(172)

Unpack

01000	00	00000	00000	Alarm exit (not used)
01001	45	00000	30000	Normal exit
01002	45	00000	01040	Entrance for unpack in T
01003	16	00006	01017	Entrance for unpack in S
01004	16	00006	01024	Set up (01004–01005)
01005	16	00006	01026	
01006	11	00011	00014	Keyword of $\bar{x}$ $\longrightarrow$ 14
01007	11	00013	00016	3+ number of non-zero elements of $\bar{x}$ $\longrightarrow$ 16(u)
01010	31	00005	00017	Set up instructions and constants (01010–01022)
01011	15	20000	01023	
01012	15	20000	01026	
01013	21	01026	01051	
01014	31	00010	00017	
01015	35	01046	01016	
01016	00	00000	00000	
01017	11	01047	00000	
01020	11	00010	01055	
01021	23	01055	01053	
01022	11	01050	01056	
01023	11	00000	10000	Q-word $\longrightarrow$ Q
01024	11	10000	00000	Q-word $\longrightarrow$ S (or T)

PX 71900-10-(172)

Unpack

01025	44	01026	01031	$\bar{x}$ (unpacked) ⟶ S (or T)
01026	11	00000	00000	
01027	21	01026	01052	
01030	45	00000	01032	
01031	21	01026	01053	
01032	41	01055	01034	
01033	45	00000	01001	
01034	41	01056	01025	
01035	21	01023	01054	
01036	21	01024	01053	
01037	45	00000	01022	
01040	16	00007	01017	Set up for unpack in T
01041	16	00007	01024	
01042	16	00007	01026	
01043	11	00011	00015	Keyword of $\bar{x}$ ⟶ 15
01044	11	00013	00017	3+ number of non-zero elements of $\bar{x}$ ⟶ M(u)
01045	45	00000	01010	
01046	75	10003	01020	
01047	00	00000	00000	Constants and temporary storage
01050	00	00000	00043	
01051	00	00003	00003	
01052	00	00001	00001	
01053	00	00000	00001	
01054	00	00001	00000	
01055	00	00000	00000	
01056	00	00000	00000	

Vector Sum

01000	00	00000	00000	Alarm exit (not used)
01001	45	00000	30000	Normal exit
01002	31	00010	00017	Entrance
01003	35	01134	20000	
01004	15	20000	01063	
01005	15	20000	01077	
01006	15	20000	01126	
01007	11	01133	01143	
01010	35	01135	20000	
01011	15	20000	01044	
01012	15	20000	01065	
01013	15	20000	01073	Set up constants and instructions
01014	16	00006	01045	
01015	31	00007	00017	
01016	15	20000	01045	
01017	21	01045	01136	
01020	15	01045	01055	
01021	16	00007	01066	
01022	31	00006	00017	
01023	15	20000	01066	
01024	15	20000	01074	
01025	16	00005	01074	
01026	21	01066	01136	
01027	15	01066	01052	
01030	21	01074	01136	

PX 71900-10-(172)

Vector Sum

01031	15	01074	01127	
01032	11	01133	01144	
01033	16	00014	01144	
01034	11	01133	01145	
01035	16	00015	01145	
01036	11	01145	20000	
01037	42	01144	01046	
01040	43	01144	01130	
01041	36	01144	01143	
01042	42	01137	01053	
01043	11	01145	00014	If the difference of the characteristics
01044	75	00000	01001	exceeds $31_{(10)}$, transmit the larger characteristic
01045	11	00000	00000	to 00014; and transmit the corresponding mantissae
01046	11	01144	20000	to S. Then go to the exit
01047	36	01145	01143	
01050	42	01137	01056	
01051	11	01144	00014	
01052	45	00000	01001	
01053	11	01145	00014	
01054	15	01052	01064	If the characteristics differ by less than
01055	45	00000	01060	$32_{(10)}$, shift the mantissae of the smaller
01056	11	01144	00014	until the characteristics are equal; transmit
01057	15	01055	01064	the larger characteristic to 00014.
01060	11	01140	20000	
01061	36	01143	01143	
01062	16	01143	01064	
01063	75	00000	01065	

Vector Sum

01064	54	00000	00000	
01065	75	00000	01067	Add corresponding mantissae and transmit
01066	21	00000	00000	to S.
01067	31	01074	00017	
01070	15	20000	01076	
01071	15	20000	01100	
01072	21	01100	01141	
01073	75	00000	01075	
01074	12	00000	00000	
01075	11	01133	01146	
01076	11	00000	20000	
01077	75	00000	01110	
01100	42	00000	01101	Find the numerically largest mantissae of
01101	55	10000	00017	the sum, and transmit to the accumulator
01102	15	01077	01146	
01103	23	01146	10000	
01104	21	01076	01146	
01105	23	01077	01146	
01106	21	01100	01146	
01107	45	00000	01076	Calculate the characteristic of the sum
01110	47	01113	01111	and transmit it to 00014
01111	11	01133	00014	
01112	45	00000	01001	
01113	74	20000	01143	
01114	11	01140	20000	

PX 71900-10-(172)

Vector Sum

01115	36	01143	01143	
01116	36	01142	01143	
01117	46	01122	01120	
01120	16	01143	01127	
01121	45	00000	01125	
01122	11	01140	20000	
01123	35	01143	10000	
01124	16	10000	01127	
01125	23	00014	01143	
01126	75	00000	01001	Shift the mantissae of the sum so that the
01127	54	00000	00000	numerically largest has $32_{(10)}$ binary digits.
01130	11	01145	00014	
01131	45	00000	01065	
01132	11	00014	10000	
01133	00	00000	00000	Constants and temporary storage (01133–01146)
01134	00	20000	00000	
01135	00	10000	00000	
01136	00	00003	00003	
01137	00	00000	00040	
01140	00	00000	00110	
01141	00	00001	00000	
01142	00	00000	00003	
01143	00	00000	00000	
01144	00	00000	00000	
01145	00	00000	00000	
01146	00	00000	00000	

Roll-Off

01000	00	00000	00000	Alarm exit (not used)
01001	45	00000	30000	Normal exit
01002	16	00005	01017	Entrance
01003	15	20000	01023	Set up instructions (01002–01004)
01004	11	01023	01006	
01005	75	30002	01007	Keyword of $\bar{x}$ ⟶ 11
01006	11	00000	00011	Keyword of next vector ⟶ 12
01007	15	00011	01017	Starting address of $\bar{x}$ ⟶ 1017(u)
01010	11	01022	00013	
01011	15	00012	00013	Starting address of next vector ⟶ 13(u)
01012	11	01022	01021	
01013	15	00011	01021	Starting address of $\bar{x}$ ⟶ 1021(u)
01014	23	00013	01021	3+number of non-zero elements of x ⟶ 13(u
01015	35	01020	01016	
01016	00	00000	00000	$\bar{x}$ — R (01016–01017)
01017	11	00000	00000	
01020	75	30000	01001	
01021	00	00000	00000	
01022	00	00000	00000	Constants and temporary storage (01021–01023)
01023	11	00000	00011	

Scalar Product

01000	00	00000	00000	Alarm exit (not used)
01001	45	00000	30000	Normal exit
01002	31	00007	00017	Entrance
01003	21	20000	00007	
01004	15	20000	01037	
01005	16	20000	01040	
01006	15	20000	01044	
01007	21	20000	01135	
01010	35	01136	01055	
01011	15	20000	01075	
01012	31	00005	00017	
01013	21	20000	00005	
01014	16	20000	01036	
01015	15	20000	01040	
01016	21	20000	01135	
01017	35	01136	01056	Set up instructions and constants
01020	16	20000	01075	
01021	31	00006	00017	
01022	15	20000	01036	
01023	31	00010	00017	
01024	35	01124	01035	
01025	11	01125	01142	
01026	11	01125	01143	
01027	11	01130	01145	

PX 71900-10-(172)

Scalar Product

01030	11	01125	01151	
01031	11	01127	01146	
01032	11	00010	20000	
01033	36	01133	01147	
01034	11	01131	01150	
01035	00	00000	00000	$\bar{x} \longrightarrow R$
01036	11	00000	00000	
01037	11	00000	10000	
01040	51	00000	00000	
01041	21	01037	01132	Logical product of Q-words of $\bar{x}$ and $\bar{y} \longrightarrow T_1, T_2, T_3$
01042	21	01040	01134	
01043	41	01150	01037	
01044	31	00000	00000	
01045	47	01052	01046	
01046	23	01147	01137	
01047	46	01072	01050	
01050	21	01044	01132	
01051	45	00000	01044	
01052	15	01044	01053	
01053	11	00000	10000	
01054	44	01055	01067	Non-zero $x_i \longrightarrow R$
01055	00	00000	00000	Non-zero $y_i \longrightarrow T$
01056	00	00000	00000	
01057	21	01055	01134	
01060	21	01056	01134	
01061	21	01145	01132	
01062	41	01147	01064	

PX 71900-10-(172)

Scalar Product

01063	45	00000	01072	
01064	41	01146	01054	
01065	11	01127	01146	
01066	45	00000	01050	
01067	21	01055	01132	
01070	21	01056	01132	
01071	45	00000	01062	
01072	15	01145	01074	
01073	11	01125	20000	
01074	75	30000	01076	
01075	72	00000	00000	
01076	47	01102	01077	
01077	11	01125	01151	
01100	11	01125	01152	
01101	45	00000	01121	
01102	43	20000	01107	
01103	74	20000	01151	
01104	11	20000	01152	
01105	54	01152	00105	$\bar{x}\cdot\bar{y}$ mantissa → A
01106	45	00000	01113	$\bar{x}\cdot\bar{y}$ characteristic → Q
01107	74	20000	01151	
01110	11	20000	01152	
01111	54	01152	00105	
01112	23	01151	01126	
01113	21	01151	01140	
01114	16	00014	01142	

Scalar Product

01115	16	00015	01143	
01116	21	01142	01143	
01117	21	01151	01142	
01120	23	01151	01141	
01121	11	01152	20000	
01122	11	01151	10000	
01123	45	00000	01001	
01124	75	30003	01037	
01125	00	00000	00000	Constants
01126	00	00000	00110	
01127	00	00000	00043	
01130	00	30000	00000	
01131	00	00000	00002	
01132	00	00001	00000	
01133	00	00000	00001	
01134	00	00001	00001	
01135	00	00003	00003	
01136	11	00000	00000	
01137	00	00000	00044	
01140	00	00000	00003	
01141	00	00000	40000	
01142	00	00000	00000	Temporary storage
01143	00	00000	00000	
01144	00	00000	00000	
01145	00	00000	00000	
01146	00	00000	00000	
01147	00	00000	00000	

PX 71900-10-(172)

Scalar Product

01150	00	00000	00000
01151	00	00000	00000
01152	00	00000	00000

SIGNIFICANCE PRESERVING FLOATING BINARY POINT ARITHMETIC FOR DIGITAL COMPUTERS

I. Introduction to the System

The motivation for this system was a desire to be able to preserve in some way an indication of the significance of the result obtained from a series of floating binary point arithmetic operations on a digital computer. For the sake of abbreviation this system shall be referred to as "SP". Now the ideas and theory of SP are independent of any particular computer. However, since it was designed for the ERA 1103, it will be convenient here to make frequent references to the 1103 and to use "1103 language".

Most floating point systems use a packed form of 28 bits for the mantissa of a floating point number and 8 bits for the characteristic. After each arithmetic operation the mantissa is normalized so that 28 bits are always kept in the result. Now significant bits can be easily lost by subtraction and this would make any computations depending on this result less accurate. But when 28 bits are always kept the programmer has no indications of any significance being lost. It is the aim of SP to overcome this difficulty by preserving only the significant bits in the result of each arithmetic operation.

A desirable way to accomplish this goal is to make SP a non-normalizing system and hence one must use a different representation for floating point numbers. The representation used must be capable of indicating how many significant digits there are.

A 28-8 packed form was chosen but instead of the terms "mantissa" and "characteristic", the terms "significant part" and "exponent" shall be used respectively. In SP, both the significant part and the exponent are <u>integers</u>. Thus leading zeros are used in the significant part if there are less than 28 significant

PX 71900-10-(173)

bits. A number, N, in this system, is then of the form

$$N = N_s \cdot 2^{N_x}$$

where N_s and N_x are the significant part and exponent respectively. Numbers of this type shall be referred to as SP numbers.

Numerous examples will be given throughout the rest of this introduction. Octal and binary representations will be used and it will be clear in each case which of the two applies.

The number 5 written with 3, 10, and 2 significant bits will appear as follows:

$$101 \times 10^{0},$$
$$1{,}010{,}000{,}000 \times 10^{-111},$$
$$11 \times 10^{1}.$$

Note that in the last case, 101 was rounded to 11 and the exponent adjusted accordingly. These numbers would appear in packed form in the 1103 as

0, 000, 000, 000, 000, 000, 000, 000, 000, 101; 00, 000, 000

0, 000, 000, 000, 000, 000, 001, 010, 000, 000; 11, 111, 000

0, 000, 000, 000, 000, 000, 000, 000, 000, 011; 00, 000, 001

respectively (recall that the 1103 uses the one's complement system) where the semi-colon marks the 28-8 split.

Consider the SP number 1011×10^{110}. This can be thought of as being the number 1011,0XXXXX.XXX...

where the X's denote uncertain bits and the comma marks the end of the significance. Note that one bit after the comma is a zero. If this were a one the significant part would have been rounded off to 1100. An analogous interpretation can be given to SP numbers with negative exponents.

An SP number, $N = N_s \cdot 2^{N_x}$ satisfies the following inequality:

$$(N_s - 1/2) \cdot 2^{N_x} \leqq N < (N_s + 1/2) \cdot 2^{N_x}$$

Now suppose $N_s = 0$. This inequality implies

$$-2^{N_x-1} \leqq N < 2^{N_x-1}$$

SP can have many zeros instead of a unique zero. Note that the number 0×10^{111} could be interpreted as the number $\pm$ 0XXXXXX.XXX..., where the X's are uncertain and there are no significant bits, i.e., as a number with order of magnitude only. Having these zeros in the system is an advantage. Such zeros can easily be generated by subtraction and an SP zero for an answer does not give a false indication of significance.

Let $N = 3764521 \times 2^{17}$, $M = 754321062 \times 2^{25}$.

Note that M is a full significance number. Suppose N and M are to be added. To do this they must be adjusted so that their exponents are equal. Then the addition is performed as follows:

M:	754321062	00	$x2^{17}$
N:	37645	21	$x2^{17}$
N+M:	754360727	21	$x2^{17}$

The vertical line indicates where the significance of the answer stops. Rounding and truncating here gives

$$N + M = 754360727 \times 2^{25}.$$

PX 71900-10-(173)

Note that M was shifted left (25 - 17) = 6 binary places and the 6 bits were discarded. If the exponents of two numbers to be added differ by more than 28 then the larger number can be taken for the answer since there is no point in left shifting a number more than 28 places. Thus adding 0×2^{40} and 10×2^{2} gives 0×2^{40} for the answer. Take $N = 0 \times 2^{27}$, $M = 1234 \times 2^{23}$ and follow the same method as above:

N:	00, 000, 000, 000	000, 0	$\times 2^{23}$
M:	101, 001	110, 0	$\times 2^{23}$
N+M:	101, 001	110, 0	$\times 2^{23}$

Rounding and truncating gives $110{,}010 \times 10^{10111}$, i.e., 52×2^{27} for N + M.

Now subtraction reduces immediately to addition. Consider the following two problems, however, which are examples of how an SP zero can be generated and how significance can be lost.

(a) $(54 \times 2^{57}) - (1 \times 2^{64})$

- 1	00000	$\times 2^{57}$
+ 1	01100	$\times 2^{57}$
0	01100	$\times 2^{57}$

Answer: 0×2^{64}.

(b) $(-1073 \times 2^{-152}) - (-43 \times 2^{-146})$

+	100011	0000	$\times 2^{-152}$
-	100011	1011	$\times 2^{152}$
	-0	1011	$\times 2^{-152}$

Answer: -1×2^{-146} (Note: round off makes significant part different from zero).

Suppose that N and M as given in the first example of addition are to be multiplied ($N = 3764521 \times 2^{17}$, $M = 754321062 \times 2^{25}$).

$$(3764521)\ (754321062) = 3646563 \;\Big|\; 460351722.$$

Only 7 octal digits can be kept in the final answer. Therefore the above product must be rounded and truncated where the vertical line is. Then the significant part of the answer is 3646564. Now 33 bits were discarded. This means the exponent of the answer is

$$17 + 25 + 33 = 77,$$

i.e., $N \cdot M = 3646564 \times 2^{77}$.

Consider the problem of multiplying 2 SP zeros, say $N = 0 \times 2^{10}$ and $M = 0 \times 2^{23}$. By the inequality given above,

$$|N| \leq 2^{7}, \quad |M| \leq 2^{22}.$$

Hence the product should be such that

$$|N \cdot M| \leq 2^{31}.$$

In view of this, 0×2^{32} could be given as the product. However, it turns out that the algorithms and the coding for the 1103 are simplified if the answer 0×2^{33} (obtained by simply adding N_x and M_x) is given. The decision was made to give this answer rather than the first one. This does not differ much from the other answer and the advantages outweigh the disadvantages.

One other case turns up, e.g., $(0 \times 2^{27})\ (1234 \times 2^{23})$. Since the operand with the least significance is an SP zero, the answer given must also be an SP zero. The answer here is obtained by adding the exponents and the

PX 71900-10-(173)

number of bits which must be discarded. Hence the exponent is

$$27 + 23 + 12 = 64,$$

and the answer is 0×2^{64}.

The division is perhaps more difficult. Consider $N = 3764521 \times 2^{17}$ and $M = 754321062 \times 2^{25}$. Before dividing, shift N_s and M_s left until they both have 35 bits (and one leading zero, i.e., sign bit). One then has (adjusting the exponents):

$$\frac{N\colon\ 011,\ 111,\ 110,\ 100,\ 101,\ 010,\ 001,\ 000,\ 000,\ 000,\ 000,\ 000 \quad \times 2^{0}}{M\colon\ 011,\ 110,\ 110,\ 001,\ 101,\ 000,\ 100,\ 011,\ 001,\ 000,\ 000,\ 000 \quad \times 2^{15}}\ .$$

If the division is carried out to 43 places, the result is (discarding remainder and not rounding)

01. 000, 010, 001, 011, 011, 000, 0 | 11, 111, 100, 110, 011, 0 .

The vertical line indicates where the quotient is to be rounded and truncated (keeping same number of bits as there are in N_s). Rounding here and moving the binary point to the line gives for the exponent

$$0 - 15 - 23 = - 40.$$

Hence the answer is

$$10,\ 000,\ 100,\ 010,\ 110,\ 001,\ \times 10^{-100,000},$$

i.e., 2042661×2^{-40}.

The next case to consider is $N \div M$ where $N_s = 0$ and $M_s \neq 0$. Take $N = 0 \times 2^{27}$, $M = 1234 \times 2^{23}$. Clearly an SP zero is required for the answer. Comparing this with the multiplication of the same N and M above would lead one to take

$$27 - 23 - 12 = -6$$

for the exponent (subtracting 23 and 12 instead of adding). This can be justified by the above inequalities also as follows:

$$|N| \leq 2^{26} \text{ and } |M| \leq 1234 \times 2^{23} < 2^{35}.$$

Hence $|N| \div |M| < 2^{-7}$, i.e., $N \div M = 0 \times 2^{-6}$.

Now suppose $N_s = 0$, $M_s = 0$. For example, $N = 0 \times 2^{-66}$, $M = 0 \times 2^{-42}$. SP zero divisors can be allowed providing the answer is interpreted correctly. Comparing again to multiplication, one would be lead to give

$$-66 - (-42) = -24$$

for the exponent. This case differs slightly from what the inequalities gives as did the same case in multiplication. $|N| \leq 2^{-67}$, $|M| \leq 2^{-43}$ would give approximately 2^{-24} or 0×2^{-23} for an answer. But as in multiplication, it is more convenient to take 0×2^{-24} for the answer. Note that in multiplication the exponent was given one higher than that given by the inequalities, and here it is one lower. One can not put this answer in an inequality very well since if the divisor is small compared to the dividend the result would be large, and if the relative sizes are the other way around the answer is small. Hence "approximately 2^{-24}" was used above. A reliable answer can not be given in this case.

If $N_s \neq 0$ and $M_s = 0$ the result is not quite so unpredictable. Take $N = 621 \times 2^{-77}$, $M = 0 \times 2^{124}$. Since the numerator is not zero, the number of bits in N should be added to the difference of the exponents. Hence the exponent is

$$-77 - 124 - 11 = -212$$

PX 71900-10-(173)

(it turns out that this is out of range - see part II). Thus the answer is 0×2^{-212}. One can argue this case on the basis of the inequalities similarly to the previous one. Note that here, since the divisor is not bounded away from zero, the result is not bounded above, i.e.,

$$\left| N \div M \right| \geqq 2^{-212}.$$

This is the interpretation for answers resulting from zero divisors which was referred to above.

II. SP Arithmetic Algorithms

A. General Remarks

It is intended that the discussion and examples given in Part I should serve as motivation and illustrations for the following algorithms.

A list of the notation used throughout the rest of this paper is given here for convenience.

1. Operands shall be denoted by N and M, where $N = N_s \cdot 2^{N_x}$ and $M = M_s \cdot 2^{M_x}$, the subscript s denoting the significant part and the subscript x denoting the exponent of an SP number.

2. The result of an operation shall be denoted by $R = R_s \cdot 2^{R_x}$.

3. B(X) shall denote the number of binary digits in an integer X, including <u>one and only one</u> sign bit. This convention is used on all bit counts.

4. Let $A = B(N_s)$,
   $B = B(M_s)$,
   $C = B(\min(|N_s|,|M_s|))$,
   $D = B(\max(|N_s|,|M_s|))$,

   and let E be the number of bits of the raw product, quotient, or sum of N_s and M_s. Observe that $C = \min(A,B)$ and $D = \max(A,B)$.

B. Addition Algorithm: R = N + M.

To add N and M it is necessary to adjust them so that their exponents are equal. This is accomplished by determining which number has the larger exponent and then shifting the binary point of the significant part of this number to the right so that the exponents will be equal. This adjusted significant part is

PX 71900-10-(173)

then added to the other unadjusted significant part and the sum is truncated and rounded to give the significant part of the answer. The exponent of the answer will be equal to the maximum of the two original exponents. This is the method in brief.

Without loss of generality, suppose $N_x = \max(N_x, M_x)$. Then N_s is shifted left a number of binary places equal to $(N_x - M_x)$. After adding M_s to this and rounding, the result is truncated by discarding the right most $(N_x - M_x)$ bits. This number becomes R_s, the lowest order bit is discarded (after the round off as above), and $R_x = N_x + 1$. (A carry of one bit can produce a 29 bit sum). In this case if $N_x = 127$ then $R_x + 1 = 128$ and is out of range. When this happens an alarm is given and computation stops.

Now there is one exception to the above outlined procedure. Namely, when $28 < |N_x - M_x|$. In this case R is taken as the number with the highest exponent. The preceeding method would give this result anyway. This exception is necessary since the computer used imposes a limit of how much a number can be shifted (the exponents could differ as much as 254 and the shifting limit is much less than this).

Note that this algorithm covers all possible SP numbers,including zeros.

C. Subtraction Algorithm: $R = N - M$.

To subtract M from N it is necessary only to change the sign of M and then add (change the sign of M by replacing M_s with $-M_s$). The nature of the 1103 puts one restriction on this (see part IV, subtract control routine).

D. Multiplication Algorithm: $R = N \cdot M$.

Briefly, multiplication is accomplished by adding the exponents, multiplying the significant parts, rounding and truncating and then compensating for the truncation by adjusting the sum of the exponents.

In order to determine the number of bits to keep it is necessary to count the bits of the operands. The smaller count gives the number to keep (this number has been denoted by C, see part A above).

Two distinct cases arise in multiplication.

Case 1. $N_s \neq 0$, $M_s \neq 0$. In this case N_s and M_s are multiplied and the product is rounded and truncated just to the right of the first (high order) C bits. This becomes R_s. R_x is given by the following:

$$R_x = N_x + M_x + (E - C).$$

Note that (E-C) is the number of bits that are discarded.

Case 2. $N_s = 0$ or $M_s = 0$. If one of the two operands is zero, i. e. has no significance, then the round off rules require that the product has no significance, i.e. $R_s = 0$. Hence it is only necessary to compute the exponent, R_x. Now if both $N_s = 0$ and $M_s = 0$ then $R_x = N_x + M_x$. Suppose one of the two is not zero. Then the number of bits it has, minus one, (to account for the sign bit) must be added to $N_x + M_x$. Using the conventions

PX 71900-10-(173)

adopted, R_x is given by

$$R_x = N_x + M_x + (D - 1).$$

Note that if $N_s = 0$ and $M_s = 0$ then $D = 1$, and this formula agrees with the above statement. Thus this equation is used for both possibilities in Case 2.

E. Division Algorithm: $R = N \div M$.

Consider first the case $N_s \neq 0$ and $M_s \neq 0$. Before dividing N_s by M_s, they are both scaled up to full 36 bit word length and N_s is shifted left 34 places in the accumulator. This assures a 35 or 36 bit quotient, and prevents a zero quotient or a divide overflow. This quotient is then rounded and truncated the same as in multiplication to give R_s. When computing R_x the scaling up of N_s and M_s, shifting of N_s, and the number of bits discarded from the quotient must be taken into account. C bits are kept as in multiplication, hence $(E - C)$ bits are discarded. This must be added to $N_x - M_x$. Subtracting 34 accounts for shifting N_s and adding $(A - B)$ accounts for the initial shifting of N_s and M_s. Thus R_x is given by

$$R_x = N_x - M_x - 34 + (A-B) + (E-C).$$

Now suppose $N_s = 0$ and $M \neq 0$. If the above algorithm for dividing N_s by M_s is followed, the quotient will be zero. However, the method used for counting the bits of the quotient will give a count of 35 (see Part IV). Now $C = 1$, hence $(E - C) = 34$ and this will cancel the -34 in the above expression for R_x. Since $A = 1$ the formula reduces to

$$R_x = N_x - M_x + (1 - B)$$

which is as it should be ($B \neq 1$ since $M_s \neq 0$). Thus the algorithm will give R correctly for all cases except when $M_s = 0$.

Suppose $M_s = 0$. Clearly the quotient must also be a zero by the round off rules used. Hence in this case it is necessary only to compute R_x. If $N_s = 0$, $R_x = N_x - M_x$. But if $N_s \neq 0$ then R_x must be adjusted to compensate for this. This adjustment is taken care of by adding $(A - 1)$ to $(N_x - M_x)$. If $N_s = 0$ then $A = 1$, hence one can say that for either case

$$R_x = N_x - M_x + (A - 1).$$

F. Comparison Algorithm

Two SP numbers are defined as equal if their difference is an SP zero (see the two examples on subtraction). Hence to compare N and M it suffices to subtract M from N and test the difference. If $N - M$ is a positive SP number (not a zero) then $N > M$. If $N - M$ is a negative SP number, $N < M$, and if $M - N$ is a zero then $N = M$.

Note that under this definition of equality all SP zeros are equal.

G. Notes:

In multiplication and division it is possible for R_x to get out of range in two ways, i.e., $R_x > 127$ or $R_x < -127$. If $R_x > 127$ then an alarm is given as mentioned in part B above. If $R_x < -127$, low order bits of R_s are discarded and the exponent is increased to -127. If $R_x + 127 + B(R_s) < 2$, i.e., if all the bits of R_s are discarded and the exponent will still be out of range, then $0 \cdot 2^{-127}$ is given for the result.

In all four arithmetic operations the truncating point is determined before rounding. Thus it is possible for the round

PX 71900-10-(173)

off to give a carry of one bit. If this extra bit makes $B(R_s) = 29$ then one low order bit is discarded and R_x is increased by one. This necessitates testing R_x again to see if it is still within range. (R_x is always tested before any truncating or rounding is done).

III. How to use the SP Interpretative System

A. Introduction.

There are 7 possible SP operations. These are all performed via the interpret instruction. The general format of such an IP order is IP X AAAA Y BBBB, where X and Y are operation codes and AAAA, BBBB are MC Addresses. Four of the 7 SP operations are single address instructions, i.e., the u and v part of the IP order are each a complete instruction. This necessitates a no-op code to fill in space if the programer happens to have an odd number of instructions to perform in a given group of IP orders. This no-op instruction makes a total of 8 permissable psuedo operations, one corresponding to each of the octal digits 0 through 7 (X and Y above are single octal digits). Two of the SP operations are of such a nature as to require both the u and v part of an IP order and hence are double address instructions. There is one quadruple address instruction, i.e., one which uses two 1103 words.

The 8 possible instructions and their codes are as follows:

CODE	INSTRUCTION	TYPE
0	No-op.	single address
1	Add	" "
2	Subtract	" "
3	Multiply	" "
4	Divide	" "
5	Compare	quadruple address
6	Repeat Multiply Add	double address
	Polynomial Multiply	" "

PX 71900-10-(173)

These instructions operate on packed operands (see part I for method of packing). A psuedo accumulator is used for the arithmetic operations, e.g., operation 1 adds the packed operand to the packed SP number contained in the psuedo accumulator, leaving the result in the psuedo accumulator. The other operations are of a similar nature. This psuedo accumulator shall henceforth be denoted by fa. The initial loading of fa and getting the answers out of fa is left up to the programer. Furthermore the contents of the 1103 A-register and Q-register are not saved. This is not inconvenient since the 1103 arithmetic operations use these registers.

B. Each of the 8 SP operations will now be described individually. Part C describes the alarms. See part V for address data.

0. No - op.

The code for this is 0 ZZZZ where this 5 digit number can be in either the u part or v part of the IP order. The Z's can be anything since they are ignored.

All this order does is to jump to the next instruction, which could be the v part of the same IP order or the 1103 instruction immediately following the IP order (this could be another IP order).

1. Add

The code for this is 1 AAAA where this 5 digit number can be in either the u part or the v part of the IP instruction. AAAA must be the MC address of the packed operand to be added to fa.

This instruction adds the SP number in AAAA to the number which is already in fa, leaving the sum in fa.

2. Subtract

The 5 digit code for subtraction is 2 AAAA and this too can be either in the u part or v part of the IP order; AAAA is the MC address of the operand which is subtracted from the operand already in fa. The difference is left in fa.

3. Multiplication.

The code here is 3 AAAA in u or v part of IP order. This instruction multiplies fa by the SP number in AAAA leaving the product in fa.

4. Division

The number 4 AAAA in the u or v part of an IP order causes the SP number in fa to be divided by the SP number in AAAA. The quotient is left in fa.

5. Compare

This is a quadruple address instruction - a 3 way jump - and requires 2 consecutive 1103 words. The format of the code is as follows:

1st word: IP 5 AAAA XXXXX

2nd word: 00 YYYYY ZZZZZ

This operation compares the SP number in MC address AAAA with the SP number in the psuedo accumulator, fa. It then causes a manual jump to one of the addresses XXXXY, YYYYY, or ZZZZZ as follows: let the SP number in fa be denoted by N and the SP number in

PX 71900-10-(173)

AAAA by M, then if M > N jump is to XXXXX, if M = N jump is to YYYYY, and if M < N jump is to ZZZZZ. Note that these can be either MC or MD address. The 2 digits 00 can be anything since they are ignored.

By choosing the jump addresses correctly one can use order 5 for an equality jump, zero jump (if fa contains a zero), threshold jump, or a three way jump.

The final contents of fa is N - M.

6. Repeat Multiply Add

Format of code is IP 6 AAAA J BBBB. The number, n, of times the operation is to be done must be stored in the MC address denoted by ws16 (work space 16). The J has the same function as the j in the usual 1103 repeat instruction. Thus if J = 3, this operation multiplies the SP numbers in AAAA and BBBB and adds the product to fa, then multiplies the two numbers in AAAA + 1 and BBBB + 1 and adds product to fa, etc. This is done a total of n times. Hence n operands must be stored in consecutive addresses starting at AAAA and n more starting at BBBB. If J = 1 then only the BBBB addresses are increased by 1 and if J = 2 only the AAAA. If J = 0 the numbers in AAAA and BBBB are multiplied and added to fa n times.

The first bit of J is ignored, so that J = 4, 5, 6, 7 will cause the same operation to be performed as J = 0, 1, 2, 3 respectively. Now if $n \leqq 0$ the number in fa is left unchanged and the next instruction after the I^P is performed. Furthermore the index n stored in ws16 is preserved and can be used again.

7. Polynomial Multiply

This SP IP instruction evaluates a polynomial of the form $P(X) = A_n X^n + A_{n-1}X^{n-1} + \dots + A_1X + A_o$.

The format of the IP order is IP 7 XXXX 0 XXXX where XXXX is the address of X and AAAA is the address of A_n. A_{n-1}, A_{n-2}, ..., A_o must be stored in consecutive addresses following A_n. The octal digit 0 is ignored. The degree of the polynomial, n, must be stored in the MC address denoted by ws17. Note that n is the <u>degree</u> of the polynomial and hence n + 1 coefficients are required. The only restriction on n is the amount of space available in MC for the n + 1 coefficients. This n is saved too as is the one for the multiply add operation and hence can be used as many times as desired. P(X) is in fa after the evaluation is completed. If n = 0 (or if a negative integer is given for n) then the first coefficient is left in fa. Thus evaluation of a zero degree polynomial is legal. Note that the contents of fa prior to the start of evaluating P(X) is lost.

C. Alarms

As mentioned previously, there are two alarms in the SP interpretive system. These are given if the exponent gets out of range or if one has an SP zero for a divisor.

If at any time the exponent of the result of an arithmetic operation exceeds 127, the typewriter prints mx xp (maximum exponent) and the address of the IP order being performed. This can happen also in the middle of the repeat multiply add or the polynomial multiply operation as well as during the simple

PX 71900-10-(173)

arithmetic operations. After the print out the computer stops on a manual stop. Starting will cause the alarm print out to be repeated.

When the SP zero divisor occurs the typewriter prints out div = 0 and the address of the IP instruction being performed. The computer stops on a manual stop. Starting the computer will cause the division operation to be completed and the program continues in the normal manner.

V. Miscellaneous Notes

1. If an input SP number is supposed to be zero use $0x2^{-177}$ and not an 1103 zero.

2. It is possible to gain significance in all arithmetic operations because of round off and carrys.

3. If a programmer gets an mx-mp alarm he can reset PAK and proceed only if further computations do not depend on the one which gave the alarm. The exponent which gave the alarm is lost but some information can be obtained by examining the contents of the working space (see coding, part IV).

4. Every 1103 number except all ones is a legal SP number.

5. When setting up od2 with the MC address of the operand, 00-07777-00000 is used for a mask. Thus an MC address must not exceed 1777 if an 1103 is used. This mask is used for two reasons:

(a) SP can be used in an 1103A

(b) If for the 1103 the programmer has an illegal address he will get an SCC fault instead of the computations being done wrong with no warning.

6. ws and fa must be assigned equal regional values.

7. All regional assignments should be in MC except pa, ea, da, and pc which can be in MD to save space (pc can be assigned to any place where the flex codes for 0 through 7 are stored in consecutive order).

PX 71900-10-(173)

IV. Description of Interpret System

SP is coded using Reco II. There are 18 regions in the system not counting constants and working space. A brief description of each of the 18 parts of the program will be given. The coding for these follows.

IP: Any interpret instruction causes control to be turned over to IP. This routine sets up the SP exit, locates and stores the interpret order and drives OD.

OD: OD decodes the SP operation code in the u and/or v part of the interpret order. It then causes a jump to the control routine corresponding to this operation code (or in case of no-op jumps back to IP). Note that for operations 5, 6, and 7 OD decodes only the u-op since these are not single address instructions.

CA: Add algorithm control.

CS: Subtract algorithm control.

CM: Multiply algorithm control.

CD: Divide algorithm control.

CC: Comparison algorithm control.

CC compares N and M and causes a jump to the specified destination for each of the three possible cases: $M > N$, $M = N$, $M < N$. Note that this is the only control routine that does not exit via the SP EXIT.

CN: CN controls the repeat multiply add: locates the N_1 and M_1 and does the prescribed multiply add operation.

CP: CP controls the polynomial multiply or polynomial evaluation: locates coefficients and X value and computes $\sum_{k=0}^{n} A_k X^k$

KU: KU clears working space and unpacks the SP numbers.

MD: MD counts the bits of N_s and M_s and lines them up. This routine is used only for multiplication and division.

GA: GA is the basic add algorithm routine.

GM: GM performs the main part of the multiply algorithm.

GD: GD performs the main part of the divide algorithm. GD jumps to an alarm when a divisor which is zero is encountered.

RP: RP is the round and pack routine. It is entered immediately after the completion of the GA, GM, or GD routines. When RP is entered the exponent determined in GA, GM, or GD, is in the 1103 accumulator and the significant bits of the answer are in ws14. From GM and GD, ws14 contains 36 bits of the rough answer, lined up. (If the answer is an SP zero, (ws14) = 0). RP then rounds and truncates this at the proper place. From GA, ws14 contains the significant bits of the answer and one extra for round off, except in the case when the larger number is taken for the sum, then it contains only the significant part of the answer (these bits are in the low order part of ws14, the higher order bits being sign bits). RP puts the significant part of the answer in A left and r[illegible] on [illegible]. Then the answer is packed up and put in fa. RP [illegible] greater than 127. [illegible]

PX 71900-10-(173)

PA: PA prints out the address of the IP instruction in case there is an alarm.

EA: Exponent alarm. EA prints out "mx xp"

DA: Divide alarm. DA prints out "div = 0".

Note: In the coding annotations that follow, octal numbers are used almost exclusively.

ws0	Packed answer (ws = fa = psuedo accumulator)
1	N_s
2	M_s
3	N_x
4	M_x
5	ga: max (N_x, M_x), md and gd: lined up N_s
6	ga: $\lvert N_x - M_x \rvert$, md and gd: lined up M_s
7	gm: bits of product (E)
10	$B(N_s)$
11	$B(M_s)$
12	no. bits to keep in rp: B(min ($\lvert N_s \rvert$, $\lvert M_s \rvert$) in md, gm, gd; 43 or 44 in ga
13	md, gm, gd: B(max ($\lvert N_s \rvert$, $\lvert M_s \rvert$)
14	significant part of answer (+ some bits to be discarded)
15	IP order for decoding
16	"n" for repeat multiply add
17	"n" for repeat polynomial multiply
20	temporary storage for (fx) in multiply add
21	various indexes

fx	"psuedo x - register" —used to store SP operands.

PX 71900-10-(173)

Prepared by Pfc. Richard E. Adler
Flight Simulation Branch, EML
White Sands Proving Ground
New Mexico

A METHOD FOR GENERATING RANDOM NUMBERS ON THE ERA 1103

DISCUSSION

The number 3^{15} was taken as a starting point. This number was squared, and from the square the fourth digit through the eleventh were removed and designated as the first random number. The first random number was multiplied by 3^{15} and the fourth through the eleventh digits of the product became the second random number. This process was continued for as many random numbers as needed.

If less than 8 digits come out in the result, zeros are placed before the number until 8 digits are reached. When the random numbers are used in a problem on the 1103, they are stored on the drum and used as needed or put on magnetic tape.

Listings at the end of this report are from IBM cards containing 7 sets of 8 digits each.

The table of random numbers attached can be used for as many digits as needed. The digits within the 8 digit groups have been tested for randomness also.

RESULTS OF TESTS

One means of checking the randomness of these 8 digit numbers was the Chi Square test which is a test to determine the goodness of fit of the actual data to the theoretical distribution. The test involves the calculation of χ^2 (Chi Square). $\chi^2 = \sum \frac{(f_o - f)^2}{f}$ where f_o = actual frequencies, f = theoretical frequencies. By reference to a set of χ^2 tables Chi Square may be evaluated. A program was written and this test was applied to all of the 16,000 8-digit groups. The results are found in Table 5.

In order to check the randomness of the individual digits within the groups, the following tests were applied: poker test, even-odd test, frequencies of ordered pairs and the Chi Square test to individual digits.

TABLE I

POKER TEST

500 Groups of 5

CLASS	SYMBOL	f_o	f	$(f_o - f)$	$(f_o - f)^2$	$\frac{(f_o - f)^2}{f}$
Busts	abcde	175	151	24	576	3.8
Pairs	aabcd	230	252	– 22	484	1.9
2 Pairs	aabbc	61	54	7	49	.9
3's	aaabc	28	36	– 8	64	1.8
Full House	aaabb	4	4	0	0	0
4's	aaaab	2	2	0	0	0
5's	aaaaa	0	0	0	0	0

$\chi^2 = 8.4$

$P = .30$

The value of X^2 indicates a value for P of .30. It shows that there are 30 chances out of 100 that the fit obtained would be as bad or worse than the one shown.

PX 71900-10-(174)

RESULTS OF TESTS (Cont'd)

Statistically, when a value for P is between .05 and .95, the data is considered satisfactory. The nearer to .95 the higher is the level of randomness for the numbers generated.

TABLE 2(a)

FREQUENCIES OF 4000 DIGITS

DIGIT	f_o	f	$(f_o - f)$	$(f_o - f)^2$	$\frac{(f_o - f)^2}{f}$
0	388	400	– 12	144	.36
1	412	400	12	144	.36
2	421	400	21	441	1.10
3	411	400	11	121	.30
4	420	400	20	400	1.00
5	406	400	6	36	.09
6	368	400	– 32	1024	2.56
7	397	400	– 3	9	.02
8	393	400	– 7	49	.12
9	384	400	– 16	256	.64
					$\chi^2 = 6.55$
					$P = .75$

The value of X^2 indicates a value for P of .75. This shows that there are 75 chances out of 100 that the fit obtained would be as bad or worse than the one shown.

TABLE 2(b)

EVEN-ODD TEST

	ACTUAL	THEORETICAL	$(f_o - f)$	$(f_o - f)^2$	$\frac{(f_o - f)^2}{f}$
Even	1990	2000	−10	100	.05
Odd	2010	2000	10	100	.05
					$\chi^2 = .10$
					$P = .95$

The value of χ^2 indicates a value for P of .95. This shows that there are 95 chances out of 100 that the fit obtained would be as bad or worse than the one shown.

TABLE 3

FREQUENCIES OF ORDERED PAIRS OF DIGITS

χ^2	P	First Digit	Second Digit 0	1	2	3	4	5	6	7	8	9	Total
5.45	.85	0	15	20	21	18	22	15	17	14	19	22	183
8.60	.55	1	24	22	13	24	20	25	17	16	26	21	208
3.25	.97	2	20	17	14	21	20	19	20	21	24	19	195
6.25	.79	3	15	22	16	22	16	20	25	17	21	15	189
9.95	.45	4	23	28	27	19	17	23	14	22	23	23	219
10.00	.42	5	18	28	22	18	18	18	21	15	23	29	210
8.35	.60	6	19	17	19	1[illegible]	23	20	29	15	15	21	195
10.95	.35	7	15	22	21	15	23	20	25	11	13	20	185
7.85	.65	8	19	15	17	15	23	23	22	21	15	27	197
19.25	.04	9	15	14	25	25	34	26	17	23	22	17	219
		Total	183	205	196	191	216	209	207	178	201	214	
		$\chi^2 =$	6.55	10.95	10.10	7.05	12.80	5.45	9.95	9.30	8.75	9.64	
		$P =$	.76	.35	.41	.72	.24	.85	.45	.50	.58	.48	

PX 71900-10-(174)

RESULTS OF TESTS (Cont'd)

Total - Right to Left

$\chi^2 = 7.36$

$p = .68$

Total - Top to Bottom

$\chi^2 = 7.86$

$p = .65$

The values for X^2 indicate, when converted to percents, the amount of chances out of 100 that the fit obtained would be as bad or worse than the one shown.

TABLE 4

GROUP FREQUENCIES TEST

3000 - 8-digit numbers done by hand

Group	f_o	f	$(f_o - f)$	$(f_o - f)^2$	$\frac{(f_o - f)^2}{f}$
00,000,000 - 9,999,999	315	300	15	225	.7
10,000,000 - 19,999,999	304	300	4	16	0
20,000,000 - 29,999,999	345	300	45	2025	6.7
30,000,000 - 39,999,999	302	300	2	4	0
40,000,000 - 49,999,999	294	300	-6	36	.1
50,000,000 - 59,999,999	268	300	- 32	1024	3.4
60,000,000 - 69,999,999	318	300	18	324	1.0
70,000,000 - 79,999,999	275	300	- 25	625	2.0
80,000,000 - 89,999,999	284	300	- 16	256	.8
90,000,000 - 99,999,999	295	300	- 5	25	0

$\chi^2 = 12.7$

$p = .28$

The value of X^2 indicates a value for p of .28. This shows that there are 28 chances out of a 100 that the fit obtained would be as bad or worse than the one shown.

RESULTS OF TESTS (Cont'd)

TABLE 5

GROUP FREQUENCIES TEST

16,000 8-digit numbers tested by the 1103.

GROUP	f_o	f	$(f_o - f)$	$(f_o - f)^2$	$\frac{(f_o - f)^2}{f}$
00,000,000 - 9,999,999	1633	1600	33	1089	.67
10,000,000 - 19,999,999	1612	1600	12	144	.09
20,000,000 - 29,999,999	1628	1600	28	784	.49
30,000,000 - 39,999,999	1598	1600	− 2	4	.00
40,000,000 - 49,999,999	1514	1600	− 86	7396	4.62
50,000,000 - 59,999,999	1612	1600	12	144	.09
60,000,000 - 69,999,999	1627	1600	27	729	.45
70,000,000 - 79,999,999	1609	1600	9	81	.05
80,000,000 - 89,999,999	1568	1600	− 32	1024	.64
90,000,000 - 99,999,999	1599	1600	− 1	1	.00

$\chi^2 = 7.10$

$P = .72$

The value of X^2 indicates a value for p of .72. This shows that there are 72 chances out of 100 that the fit obtained would be as bad or worse than the one shown.

PX 71900-10-(174)

GAP TEST: FPequency of Length of Gaps between Successive Digits in Sample

etween uccessive:	Sample Size	Average Gap between digits	f_o	f	$(f_o-f)^2$	(χ^2) $\frac{(f_o-f)^2}{f}$	P
0	204	10.6	10.6	10	.36	.036	.85
1	218	8.9	8.9	10	1.21	.121	.72
2	209	10.7	10.7	10	.49	.049	.82
3	207	9.2	9.2	10	.64	.064	.80
4	205	9.6	9.6	10	.16	.016	.90
5	216	8.9	8.9	10	1.21	.121	.72
6	205	10.2	10.2	10	.04	.004	.95
7	204	9.5	9.5	10	.25	.005	.88
8	209	10.2	10.2	10	.04	.004	.95
9	202	9.8	9.8	10	.04	.004	.95

The Chi Square values of all digits give a satisfactory value for P in each case.

25425018	1544355	92934827	95996944	92154804	51671604	29706453
52091654	35247776	44974009	56969404	83906507	75883574	9915732
41505940	70739474	72593573	54568290	48114715	94959820	41387949
9772606	41666627	27727856	91241977	7924055	65641900	47736366
54577061	26421999	54417258	94301999	57841452	54215149	45786057
84209253	2694794	89987106	5589360	68983772	89731175	53584256
14462379	36712770	32510446	13029317	24549336	63223009	48719068
50758033	99246681	15613509	73431533	5449197	18280823	81676930
33873158	96319727	45769069	8961893	51982537	52613798	94289925
35185855	17894992	99248589	29817344	84631187	61236154	96069017
89350406	79340621	76322759	4476355	31042506	56508932	14969277
69512902	94849004	10958290	4790313	4830377	58155151	20716922
27912823	60178285	26506534	93172056	80277137	39562859	35271808
40319279	90067898	14631837	94226739	88136797	29814693	10054171
43709917	63668181	89895170	547953	57804581	90714747	74239768
59640776	72092429	49419970	38284777	51061924	58268645	68309031
50833670	79556969	67145092	56429731	68615501	82407170	41420007
69221130	74101375	58730855	93543697	66170487	19595307	16193469
14102943	2840746	55336257	43242663	72377528	16662798	63766778
45019601	69798315	70925172	95892930	66288920	89405394	15631291
36797697	92972589	16067295	88498249	69661847	54149717	84123845
15945567	19531435	98416637	34620792	56317660	2880616	21228542
59332898	909559	86031030	25077660	27683326	46498963	6775191
8157247	98755466	32700878	1487388	48733049	41413508	20611236
7580823	91731621	92211535	36995748	27330278	46786549	43981443
51733249	19051818	27769798	63180270	54359847	89848331	7939262
6274564	17552991	7653376	16892124	72635587	28532062	32091121
27883443	38976650	53023418	74853550	12256901	1558451	86825442
49104373	18389124	41355478	55804649	66547138	12210518	38905209
56535823	6343999	91826215	68942682	25526723	44101968	86857683
16572943	86133813	54007003	95874774	10753799	19180257	17092689
77716255	14247515	32030761	74687253	31667667	60799696	59521323
67125128	31680297	13449306	18840584	4739498	51715372	53013146
54320653	60998096	3696959	76574637	24462366	32532139	86418381
6311923	7195700	26611000	79183438	83689966	24024781	70573170
95020039	16734921	79440161	57975340	62609695	50836254	18031436
45329916	2892490	34585569	4960496	94739221	9588631	28506238
94023920	19647381	4726226	81005240	62821931	40020578	26762593
48778011	98309397	49685098	72409322	15959161	58497433	22649306
30891640	83659401	32819012	40179798	76699566	1480104	30278371
34963756	40785928	82487729	25976570	82575826	88371814	64563019

10-81

WS-174

30235574	38972899	81719311	18161246	73221304	29461346	1930392
81158289	73376193	59256184	22651243	65382654	23977131	72277122
10648205	50602431	16872305	51965243	60661784	72142727	78243184
73681902	69421496	73583843	74799945	84930415	23297942	24484348
85799209	4989745	61280889	40285921	30871647	81800819	29414749
33256977	76215763	81509616	24674031	81438714	43349398	15760674
78439043	80592733	1042220	59908037	74393720	95597922	7077267
50337986	93789752	8782829	73194638	52087352	48443127	72709439
92845075	8566187	60189566	75424270	95645010	21056267	44790362
83493094	96993618	78738067	36915664	56136251	96431266	96537913
9686705	41048495	86956763	56957542	18644279	68562415	65405178
99241297	93116777	1582427	90221346	18479390	30550553	41483190
82242238	99230481	57280260	7921630	47128147	35840062	31882297
58898898	5584842	76102686	40326302	37349879	57509274	12159597
82808249	50146629	79309901	17424707	22690892	95552779	79307277
77824974	39043437	54210604	52145416	71266766	41820832	70802311
77234724	23974418	4248506	63724150	53782994	34114065	28328041
64005183	40860244	25636101	34078307	96138170	38543974	94945011
27439464	31361611	40872701	45449505	11685696	38744218	90995418
51133359	54587450	50246299	64619597	11255575	18324460	43372411
98524753	68207830	23860994	26376659	63452110	51258255	78457617
66239168	72657264	5016687	64872271	5853609	13545351	9001869
31846867	1718482	25359459	36309423	47225475	43664961	6941082
6044463	25531193	35632734	69761836	37843614	49141598	6165214
24378609	92228274	87751186	53575316	31400845	70848201	78055015
14668037	50850225	8405594	15183553	57653498	1249589	42863195
515447	4862188	13527110	32723275	94681137	95020891	14585745
17541506	87214510	72233635	45025417	26766480	23908024	7231031
82412964	70197084	14667672	40582171	67637166	74859404	42721878
51566297	20036349	75403567	50106283	18052001	86986474	15798176
33928345	38208597	55476148	86293802	84283379	74715880	31686166
44336257	82290366	96849164	41539917	87044596	40253451	1196652
40393830	57976038	82245590	59310058	29471215	11126232	3570554
50730232	40375683	84341820	41785869	26759378	92552471	98630749
97260693	75806870	82370130	73165356	33642619	53327221	31399929
45079606	74797687	51351033	4101679	93987312	63545604	27794393
83989667	53082604	70644411	34207296	49449184	54816558	52982408
14866526	70011250	21341220	7925969	34973667	51163298	91514679
29071949	53402300	21445062	85890509	54567114	15031890	86089708
21219942	85836127	1106498	49651483	66873439	64791943	27795458
76605374	36057882	84500605	11627128	1742225	22200687	26963453
62157518	5247943	52334371	18174384	40161794	50585291	71566662

10-89

WS-174

PX 71900-10-(174)

22674565	70380456	72951720	63506282	98811223	74019207	28910460
73852331	32032998	3400036	85615012	89008974	3752319	63806711
94659128	38141419	44696068	22657501	59764574	65074040	44804789
17708331	82657258	97487311	8495159	75320344	94952517	42720054
254159	36373540	69004099	8458071	86155936	5274951	82508524
74103409	89096890	61055734	14536513	64134348	50606715	3170584
73201615	39574561	96032970	31883958	26354069	99046679	8195829
33214293	83403783	26287239	98056415	46422514	3621999	21568179
84424881	28869386	2671	75139648	7603252	83216092	91914267
96893230	73229843	72855633	44774575	83705637	62222751	95529684
61935511	68056984	73874183	91107404	8714289	65794263	38992313
90996335	76930086	43938108	40607562	30813014	42970268	59866052
53674185	25242751	81062189	54278775	90860229	6139530	8262661
80095769	36463914	98345582	57101315	36684772	3213271	66097990
61316019	89026840	9724529	31358363	83718815	30288508	51698880
79278983	87174315	65584858	53663277	89970185	66444549	79002251
86873582	61184931	25183198	43851841	29693245	83181948	93934924
66179507	4907373	76696357	45423353	81293040	58305751	96594491
33185172	69488246	72323576	47180056	13343621	42978511	23319822
80526018	3287095	91080222	20891066	95033421	98639917	86736244
95560836	37528560	57659263	29210848	68546838	29851583	30362324
66518645	15961660	69600964	22041913	30336502	59672216	17027544
95079140	42613740	96473208	68476206	56122497	15549417	15248009
49512882	46671028	15410079	8533705	85941267	6078615	8110311
8080848	97008857	33947181	31225525	62503966	87726895	41317942
16536122	57737650	86883185	62897751	25478958	42570780	95896892
96098816	5192675	49649912	64482686	52036765	20812761	39667879
36868126	60990949	71075482	29394591	3050002	36473676	63204790
73059370	4871123	21749834	94406878	55471207	81513026	5387095
16079292	23340522	14370636	5967242	96229049	44983753	62648788
1059766	16635131	90754667	12903592	74652819	2506379	44280364
5158619	31122389	22048496	40546150	48022325	46013112	29191075
14599627	5411947	44118525	15762572	17112982	54343379	29229432
49170550	1009419	76685740	15185036	99372741	32757560	63012345
27972279	85573467	69687478	39904749	63885487	29056951	34136933
554071	17677379	17232662	54459	55626753	28094301	65180980
62339674	29323899	27633016	87409855	43322108	25595427	97810952
64203799	88991261	42327238	89761669	54341634	48932499	15127382
88083080	78564694	66311331	89455913	94637055	88280054	33339253
91158471	74681235	99205987	39312160	96233791	44166329	18390894
91148506	62784756	67968153	95601708	20897173	54058228	60523757

WS-174

49087588	83070219	37821065	25195863	99241761	49054460	90648871
97264750	55719263	80416808	18036874	64092124	70735296	52632521
34127977	96407113	88162707	37617216	42873974	35311093	14056192
18507386	12817978	82108472	94267996	40804907	79527484	98596684
78479986	24427503	88044153	59371328	8267210	37355315	76215218
56177864	92904793	77919582	97027813	23992558	75780024	37263952
56159719	21777753	74508166	93927758	47428678	56864272	79159443
6722890	13238603	71009797	60615071	34161114	9716964	86977855
79759561	50637946	76443911	57239337	27157850	27939870	15748771
41766903	30085335	14757794	25726288	21050981	30304166	89531139
63385233	48945802	3111794	29954395	47788428	42772239	25631638
42744462	15802937	33645423	67198429	80492769	7461818	13487307
77425993	35978684	4013496	29172745	70388238	23436798	38419225
75210197	76683808	95052407	95876593	61925278	48621221	56973326
60021127	34577032	33236019	23503464	34795899	16311312	88089218
33515994	5488244	42995704	19525887	76559947	37734874	16419552
65958525	96551359	39106492	81815019	26231056	73009383	75049128
11262737	85585640	68032392	53149181	42351625	44353244	42127664
19458903	71237408	21238093	59582872	53769854	1149761	808484
61196360	78264419	19383557	99113431	66046664	25910405	20294853
99791417	61071306	49949450	35265287	67634916	11563182	81883651
77914561	89996504	1535191	47717476	60041023	57413514	95206792
9797018	57327932	58888953	91187400	83195494	72353129	1367205
75468204	97600817	82941722	46866539	12531095	64678773	65361614
81673461	70502177	7503885	65182105	6840065	88339724	8604620
67631611	52805792	51910922	51231296	89785029	92191358	6254271
55375752	80560525	34501994	69776592	45636131	27292418	55463941
11326154	79937881	28490336	66229265	62504948	65640294	2556929
27877963	19032913	32811786	34505159	24595550	21122001	14948334
17222578	21975836	9766068	54908579	91548536	42001273	46632480
4732893	123524	3960716	20779040	30564460	30056922	5902272
13345934	36216340	99655215	60372524	41696373	59225677	71613198
72514719	63885271	22980514	87297364	37495860	3446732	57073461
58412042	98065272	27149121	50814116	66341513	47250332	71844852
63991250	51555088	98289467	25892140	77693926	57184446	93595950
59738811	11406870	55881901	92290048	98212674	13061659	92153995
28913095	13761512	50170634	83521623	9088054	46898812	43354291
85688645	12943427	64910081	32187975	68192128	8506734	62427438
82351303	80401613	79406202	42176587	12449386	47770808	83963402
65821309	18258226	66945584	81084222	3014875	22009536	18278722
22572324	12744277	22043339	29688936	96180378	17965286	63469176
17403804	73824640	58345727	13225951	16339263	88997509	83876087

10-84

WS-174

18091996	4152436	45470681	70530695	4456034	91246182	54789731
60564009	5924788	10230503	38164847	32675825	40787536	62771679
60690317	69516851	71723179	11721358	63820327	75046549	72928929
93446389	13090031	34998923	71493122	554150	19899780	71743999
45401307	12039329	54913650	99986427	22218860	1329094	77084552
94354925	70342379	75524543	97918093	47509941	70953787	95573850
980725	74782420	24518251	3096729	8804207	45199522	92789943
68224558	90387541	2225549	62913840	75435960	21850864	47709570
63306764	13351941	70985460	47063640	54381440	26206502	94280016
86491620	53380232	46292046	91437711	11318429	41680995	29270281
6343780	85665383	29633525	13772289	84910016	70959535	49713251
75652060	61303598	8039018	95539458	98450154	61789435	86568476
67983007	6044714	22170668	76455525	93095260	67362261	54980387
98360448	9439173	56885700	10875891	74603670	96258710	74090383
17943167	86411323	76194163	10736795	53917196	22831475	55333307
94472010	8693922	29707771	89169174	81258313	68644495	92749987
17938003	97782728	36621539	15151354	91366457	20091454	14985641
5218902	32401370	31548737	70545251	11287210	74904816	82121673
62980781	79548839	28259521	15472423	27157685	23298147	30251337
79758982	34349718	80968099	91725983	67822997	68101461	43325467
60417199	36666189	17313651	69147492	15808535	56908696	79633104
10382052	6521628	92556418	75448661	89199648	27164823	89884167
62535014	21634402	68339231	95103262	50116865	47305711	19134999
86087559	94982745	15217484	72664564	38396641	99606066	19907127
64053591	60480085	60922937	36833602	29297812	99068918	75648549
96751088	56407410	32662741	2372312	30850252	16794987	24365644
3467241	97154647	45347474	94174892	25101531	28125406	691991
5281028	48866354	33407300	10083135	53208784	55014389	64849054
23809631	57845508	34099411	18739963	54821461	56694181	9600486
31484497	8198242	44053082	53790911	56647093	9352654	38882206
80512019	12125251	56122546	16927868	4431615	75496470	4687795
73619458	57598939	77027398	97131091	53766866	51310106	60717016
15297521	42536403	68339108	91643068	16988257	92661350	74569393
71514091	20160375	74799195	93773568	90371048	69094734	8033726
23958819	94001181	51050333	630096	8902749	34413354	8694769
53535283	33152061	77545136	32225172	40863449	97970267	38855864
10556622	85626310	4187221	84506180	34243373	42097338	64745290
57546670	83008968	59559572	69390949	19552568	21834248	53193240
20389138	67829883	32487495	38468359	51747886	28161658	80996626
88931317	64617726	92838978	46211010	93462349	59418049	82925669
22744759	31787877	87062659	11524610	70531306	21644475	83273929

10-86

56423209	19358580	67557742	71961798	14667992	49584301	97784908
58891947	78477169	79398361	90031742	53314329	71379825	1132063
37203422	98181148	65707222	6304096	43030638	13343557	8333181
82013997	6839621	75849269	77218325	99514331	17008443	12098787
85791391	84786092	84222379	72520727	18555722	30412340	45636967
50810481	98300677	72812528	40119169	98513435	18249273	13572919
13505845	47467862	17308123	98433307	45817454	27934706	4694636
62311396	39123347	19103140	79857986	923732	82087782	49094981
22612083	57488896	68343509	81232751	20631788	48871670	48737446
30890664	79909576	24294417	5469412	50091303	63074922	43536346
12667021	87187961	46816229	73622575	11067577	98338836	72546501
72532339	76785035	30942708	67598385	57627543	76398870	23852774
63569917	9920853	51350414	71901716	53056904	77346030	68188587
17987900	28549766	80745284	62161213	67039003	24760675	26748516
55421898	36546564	7507706	79703610	87256492	95658438	98879723
82902490	16947302	14271272	29265832	4724982	46009462	60728265
63315045	87707171	23320104	48430612	36945607	93176008	57235706
59229541	46096182	84386221	15041455	86733526	13492069	23301378
78826230	59160337	10414795	3174323	44168142	69393583	93651345
7479065	79291267	30090373	12034052	40680638	73295864	81169902
4910667	35144299	19485526	14880072	50759502	6354353	87026469
67181865	17173231	61794580	86013466	23890004	37169582	85730063
31783261	91424671	47134841	44117438	20166919	70914806	72715657
93275476	36862808	45604540	78109424	68888208	44131148	89004579
14331426	13833417	59711316	43233403	80313865	74870001	72397562
43381448	84315057	26346329	15525403	10783103	14460563	99986986
1907458	98638973	60179962	73683319	84054922	90293478	34405958
52479600	77174566	76460823	95956170	18856556	37653565	10764248
20909133	66417107	10256546	99708879	93634612	73623006	23192320
5952774	91873459	55817278	19168616	92265285	72677158	94828930
63764413	44080442	18166605	89761111	90427091	35064300	50682689
47004041	31752184	22483618	67434582	12980926	21115788	84987778
26401291	85302379	18277817	78437027	23879318	4990181	33522945
42142825	2192569	74655978	91374275	60328232	84014133	16573255
94910889	7055253	66655051	95191712	82862508	72134995	38912414
5565125	58301770	47763003	8333629	94616978	60351743	74328817
54474590	96533268	13232899	77580269	81050486	9129430	67455023
51149159	96414403	24807011	64289718	68937559	15625986	21776999
53296899	15103765	94786441	52755925	66025715	16153215	89650883
76538845	43535246	91211432	88166770	67698400	52642659	69183921
93309719	60650950	35799841	73137711	61249404	94084881	57564809
56418133	10780031	62257849	9152663	49949107	25616129	9103820

WS-174

38792823	29247946	38433125	63587008	88065772	45029867	17734614
16735160	86163626	87389645	83376218	56144556	99234667	40821637
13299148	62226042	53893230	48523098	40724589	86842954	4874991
60849297	8917635	14891333	24302887	67797348	17641300	76012531
1401826	87930132	53679034	85835411	89169915	91365881	3887621
76171249	37216709	69313255	686827	84218777	2185213	1937257
5846640	58938478	45293895	63303994	45259715	41288147	60574090
21084539	34822124	82975306	52115921	46829991	58117217	27312572
85558390	99873403	24346424	63492021	22718992	31077124	90846445
21025992	78197294	23493780	30804414	69473497	60062990	4292924
2935089	25006358	35106031	16685203	22967853	110775	29276260
29646417	82429939	10861004	58460916	32457606	2947712	18039862
13931779	8224808	84255259	88960129	23785000	36023371	53173180
94273247	54306495	65363318	70875156	31037389	46776985	43365758
45582585	25975619	55822623	35314580	78870359	412090	50048811
11882301	66641263	65370992	77056659	72712425	42628467	77190266
20165934	96965333	88340593	64067949	61741193	37700617	89951905
89068821	76734698	50277287	21533801	56838965	38320308	57792035
6210468	93474669	3346846	65687328	41036635	21750799	47912140
33334527	46930462	65960368	14180270	19365860	3920729	23771933
71080228	28689695	37383158	54178501	88559014	96317312	16226745
10687775	17080782	11188104	42226156	64730077	32232998	49494358
84550647	77214093	49803474	89468487	45711292	69004368	16025486
81430948	93314426	58848919	75987900	82564393	35178067	29911132
38688220	53350942	27622260	73441826	26572859	79955835	69223413
4108213	9364085	92019690	8787649	17226938	96266329	19990001
69644701	44476525	20565210	54970582	90965034	1687543	94518303
58110671	89859134	29403952	2675652	4028088	37016661	78776652
48633721	65724838	99218134	21503691	15101222	23247644	85913408
27665226	74186932	49671697	6242135	44795670	6054244	98832484
1037669	66098606	10729785	4419830	51513482	10658036	21232174
27289339	52966258	77460927	79205799	41653247	53979506	5155605
80551311	43732388	24725944	65442275	69099453	6569275	27009461
14261388	19648017	22617959	88572768	80586901	60010240	4236809
3103645	69145000	79922181	62004816	4455365	6065409	39256869
17201501	51770079	4702463	83601379	82062754	16104577	63560761
20782882	4428548	28329789	8419521	94981925	29188297	93855755
51484123	76285354	25953650	53046425	50989881	44845794	73948004
39118199	8499112	52307046	54783298	56847682	15108482	93265726
31014635	77757963	67280878	25488085	30892933	9522572	95894193
20171481	95602264	23331784	36991697	47586674	4243034	44005465

55006592	30244081	9853426	33565551	32760536	69152884	33276687
93819757	47566748	84681145	63700948	72160991	99134259	34750780
89216482	69933335	86829452	27694579	94629013	52813526	75824399
77646560	66877789	52946911	70068723	27539051	14336380	64573605
59601810	49656212	69325176	48297462	1443856	84303990	75551728
57370747	35902189	99498305	75419673	29185373	22981585	31212114
49461732	5159372	52305524	11966920	31194614	40591308	31793262
11620892	94748685	7391151	22044116	51547232	20577272	96369669
37126967	94859167	28424837	8199800	87882199	61266204	29825761
52979865	57595163	76789746	15447827	96003229	526795	55664745
23130287	5692327	12825804	33831343	64873946	52974130	30477853
9577842	65559958	24273725	94457861	13312330	30404325	88597213
97176550	9006667	13615489	44572923	48045861	37031087	81950713
85431370	23129380	85000738	76045856	78241935	38545466	2699711
26293286	94884789	78125443	41620902	83585244	30811695	5864616
27762183	37102867	87974938	39856013	94897383	29762314	12941259
60546018	36679108	9949487	51565835	7039525	62334914	29634953
53944292	76541065	91606367	58270236	37022885	49818599	18615642
9959214	26247626	57419017	7260238	97341241	60053984	19374702
44907870	75418113	56656142	52064064	64401802	58337415	47831097
35162244	69342880	9469357	69865426	79820431	18181279	99911823
31422758	16208502	34352918	36771664	31709872	40136614	69821759
59355035	86789078	99866720	23905786	78490167	42400419	980614
71659806	92832541	58617216	250149	57783334	29872400	22834809
14906515	48744479	94522861	97039395	24343525	44781411	7578170
48706578	98080479	52294003	90515315	10115550	25572262	5002968
81587290	62288498	66053367	80259156	12210334	33728985	67026572
77709264	86014576	55116140	25002802	69287597	64879849	84817557
69641173	79445776	81716850	83146973	49554411	62236129	69221771
92133765	41447128	26873358	77772956	14065200	3481872	47852853
68 35837	48644831	65472205	78947648	27195871	23791674	89556303
38243703	12028038	5714597	75834149	83494733	34724365	17207474
85582722	81041753	16251200	27558741	31377741	91983100	53192242
85565813	47136995	74030737	84832978	93856147	22818529	93794095
73976785	43468974	35765819	55566664	82521790	44644751	55420406
54844666	38400579	87538283	76726255	81197185	93843195	35827822
54983161	42179929	53756282	22000133	22192233	57573903	43811941
73652924	59528786	8636208	16730211	81157797	59535419	61015743
97483078	23631595	15454239	22087977	77470169	70763092	99717153
57960464	46776390	26627423	4318684	24681465	22133976	29320855
76217988	9884819	44071963	48073764	16682269	74647307	20310795
11395338	57774748	56769473	96661307	42203888	50235093	17811735

PX 71900-10-(174)

24001541	87882164	11466299	40759148	210667	20831183	21038302
76276105	34199436	14036396	7218551	88820586	72795349	93716392
92528318	90255543	40973476	96703202	96136960	4504672	22340192
59155773	67601134	743732	76313345	8080418	84912246	49044827
88092102	76977895	46707089	90341569	45108065	70090068	91138300
44108548	3528513	76006906	47886706	86779918	63718478	28437979
69735249	41604726	31180788	54296240	45303660	76014686	62025541
50612988	45422723	63570098	15012682	82607804	48441760	34253468
57650933	63309624	69423201	24311288	77425367	18368268	96778725
73240240	29401576	65331879	50485089	71290926	50392960	29686580
29902200	55851740	49117590	79459879	75804387	51546806	8593188
14465996	4247235	27968817	24769400	79971893	12151311	10270637
93458836	94809276	47778892	42944239	67460786	79054155	70623959
77989808	59335447	72617146	46627137	40242439	94064059	75082028
62855378	41415923	88549181	88134211	91283963	39911068	33072875
44622157	90902448	14421944	39277207	52472174	2486629	25551104
58891191	57209639	4287567	88832874	15823933	87426724	52728004
85868164	97053939	30837121	50051418	85221524	25394284	76470188
52930956	23881363	62519414	82648240	54396660	51717096	87429591
50233969	65148285	4406310	73826700	68051206	45547213	76018814
43935282	95325235	84507535	72361765	64736883	89479309	81716599
76085928	52914936	16128595	68138194	76744453	56266118	29313899
14751929	77757668	11868188	72272410	12309578	7054596	83653892
82945308	13528923	83725965	97212108	51207726	47480565	41815861
65177586	4935048	49933522	83107586	15274767	73517381	4576351
25523121	76989477	64730933	56313694	25528204	85764948	11989397
20460374	58550488	2129313	39990517	30238919	69143167	28356859
69396829	50748967	18925242	90856180	26452190	40780443	62403333
75326416	27204670	2886644	4345646	44630084	45466616	90393248
22528819	71469092	96441343	65564899	29054501	99432108	92236745
75106113	3732322	38128932	96069285	79627684	98816156	78574930
11004729	75151090	61050185	63059037	66648021	31643910	63564625
52783332	31723597	23590385	64108318	88394976	45059756	53255261
20312005	45434640	25025032	23565632	38854119	95684613	29919406
3014071	99417030	70720846	81384533	71257114	38729561	81322945
7801383	19824313	16060149	21687176	76564147	84233723	19987734
49960473	76925056	36653615	66239414	79577651	33482377	27765699
82086347	8726057	12295888	24585433	4949510	54120322	62500235
40113126	80154102	9657540	25895166	74248016	23235204	4391085
48175448	58649117	92371684	70825927	34008617	14691374	36270002
11989354	81998217	79620196	22384057	85191237	62303565	87259818

10-89

WS-174

10-90

6046535	45170937	93952973	37056823	640600	35962190	76880902
2488103	67017163	81453985	70294706	76581468	34631195	80535803
10119877	13080202	76928349	7808768	93358852	48048105	3053149
64615073	18205699	75415935	60001165	9883861	84446082	88417677
12587800	49271494	22158842	57753931	8023737	26697937	41381055
47178755	71531352	32050828	97387445	17671659	40955435	19161846
36031713	19411984	93505600	68180677	63901970	90097825	33290248
72639937	42449204	31206089	94966585	97480862	95509577	71923100
80883631	49093342	3057312	24909498	72349784	41484549	5846169
45688469	23251933	72352346	53950373	83966772	26407256	36323319
35489759	74170403	63975315	5930361	78398955	26592800	41814872
37355380	78043776	88564337	35425300	5662563	66999858	55506594
3266052	74519827	19317794	94589573	94710125	20789307	50956699
50905337	82402317	39114817	47575585	94944303	7522253	86281601
50778360	79944587	21234022	79276636	21149323	45270039	63283890
16571849	55357784	63049557	1065575	45834183	95895967	70077036
69424969	37067351	28375200	48488588	9425295	69127786	79603516
90018975	96810683	32747552	61683383	88818833	23480559	61528997
40652549	88412194	24408514	20385531	80204956	98087408	78556475
28705961	92319463	36344682	99595377	98165059	15750220	27024011
20923774	75639473	14570805	43513254	34139500	72768028	53559397
95322313	38368382	95088008	23649794	31470251	10087879	52447624
78 57036	22860385	63313065	32006495	63133170	71536209	84594936
82110463	16060392	28523168	50323643	92950619	24903492	94105296
27187505	56877710	54539429	41513408	12391213	21880818	21107100
98144693	16894715	11306976	92331486	55972612	94003709	54249565
74440975	82781966	90633586	72747950	43731429	77323303	99929120
95906744	29969380	66688083	84912421	53967866	68628044	32609326
57370517	93589604	15433061	97404442	93172204	84440621	69008441
78 09172	29357881	44078796	46935840	83034677	80820082	88512477
77421769	51368304	87310296	53247918	82176724	1043745	58298649
31021210	62198953	97141215	70136278	48924765	65992623	62942410
73850956	33116023	3260675	57473650	42101491	47358322	22774456
61907247	78250275	4728007	30409695	37606206	20654513	17074874
64602017	53572004	72446534	78867542	55382948	14560453	14136320
72159248	50100786	2117550	68574786	10768690	11652214	36364265
97552227	89859166	30304165	89503007	77291902	21873411	81171393
39360207	3003582	72779564	52311413	43416505	97114688	85526579
17161494	48809	30900634	91946714	37480200	99775937	62410690
13855541	11007396	15960484	36518138	13143669	85161749	88655113
94351476	64463033	36036276	13558994	99332837	31558865	87027189
87436656	63696138	71065462	15950871	34524002	17809400	83842639

WS-174

29815304	27242612	57398297	69780593	25626309	27048604	41676909
48542070	37567422	3338256	92471874	75882362	97628902	99370111
92989038	31951694	58822382	546830	82411527	29771897	14091446
70736631	26414089	42678106	28147077	73462159	61702874	84446263
93509506	43845191	58394033	29976364	10667987	10469421	48437028
35352208	96947335	91397644	92126988	74467876	34245230	94337820
96033771	54417413	98662406	48300377	83447628	51760070	54628416
91054997	12537131	11530505	37927816	75488075	11537020	16991952
62390085	73367271	76701021	42411661	48801216	80015795	11166941
75040245	29804708	78584335	7147481	12308041	98033971	86124305
54966690	28137741	79260491	60794796	55893945	28454508	71689786
3746492	34101290	71599341	2464800	82553862	7359071	59104742
24954785	60667232	91186252	50900355	76468144	67865075	43865659
15763260	46403767	13108788	75793451	12334491	36809859	32456584
60744638	7022246	55673438	7052720	30878908	17629305	7008794
3127414	66717651	95523153	51185255	38112773	44143067	35080991
50291873	4514330	25600254	99384720	72165264	85123090	74855313
61853008	28812485	2008061	53329840	74413571	24181494	49984085
70081037	2473821	67894060	53955291	6366282	47595228	66305549
61016416	53157558	90337274	58500338	67721858	41466366	31199211
35694795	70809969	76267432	58649054	90599390	50281995	47815191
99419358	1993609	49984342	77310872	18659981	10600470	85998700
11151180	34310426	49360962	19308589	3405492	40580671	79631067
59767815	22031003	85310033	99379770	96277230	58217708	35718356
62531236	49570740	64009491	27833682	9744379	52900789	91078599
86361794	25038737	6456770	43035530	16745882	84520066	22226341
77564272	33681596	76072887	88703217	65252313	93863259	63390422
60703229	30099588	43178031	89560340	79551302	97547786	99144217
46450338	46453714	76023245	34368914	84112423	97278996	53829627
72050397	74949828	36361754	42988068	23738586	72498048	51646574
74454182	51663540	71288248	73058656	88464766	6054973	14110380
66510461	54167657	51934811	52179745	97474231	43186305	97546491
62713725	83014537	82007027	59024614	52522966	89172334	35936613
28446220	33132756	71334462	67173109	39287922	85467394	97019998
15544739	17865859	84995472	30487132	2176007	11392703	38603705
30446390	63995699	42495090	15960025	23605709	92544604	11536753
9480800	19962521	11765714	8081555	63797973	79760852	86955912
95562653	88643776	64893489	65880138	38879022	92976259	75110468
98630827	99454962	64068932	89394609	64728616	11160266	21479181
56892003	53971980	50143941	3692014	30089731	96917609	87962803
44678457	64101849	75880178	36189369	25351050	68185798	73746444

10-91

WS-174

10-92

77642709	92760516	45700735	99879349	57399512	9375392	7451121
98486688	71118284	25526106	26744737	83330104	83559273	29012752
44283514	93773330	83675714	25745628	28246263	45156134	80173279
12267362	27408230	92221617	52183630	72547980	29758219	31959862
93447015	30700446	97471679	67336718	78476998	74587848	53812859
2992002	15450463	50080578	9590513	19362965	56697318	80087711
21017072	15911996	14647765	17436241	78725936	98303749	25015242
16592192	90769115	11964918	71741208	17377568	38770120	14355160
57735420	24149596	92164673	60867444	86336446	25077732	29708805
53016132	4104045	26329078	67097147	49834846	32490650	98217335
9965572	41974418	96463272	57395545	87710968	18102123	20606243
1337201	36712417	95916992	46805978	53680443	36007251	2344096
42393862	84956813	94828712	10520455	7712865	79803370	75098037
44976537	68304135	47318819	30475182	68655922	11557217	48296061
96249136	73193380	16697731	96594153	8337264	62657813	95366650
77681987	74920434	14766275	19888865	87184233	76159150	99505007
26365415	15575245	70743737	92099897	15717305	40594706	71805651
34944094	24532253	31520105	70386093	97312127	46335369	32332063
95094092	60585051	61000146	61366850	42591952	20411637	29675333
70685873	74902588	49599685	51315750	85274344	34913256	96532394
96847592	39715104	23208220	50592893	16988042	86613044	15245344
51106084	92602947	42209123	56917188	91379054	91310603	59438771
26502179	52562762	740096	16376978	76241027	54887798	43815540
82227886	98138162	64399482	75211299	9998494	88036537	27907322
39643783	30096424	16767316	51241006	77449441	24521074	73472564
85977281	79687917	48439626	46698915	22508552	97780611	72227957
95892703	4071653	54612628	87456451	83691323	62199436	59403410
57540311	38337506	58020916	2560692	95799896	51795436	23644539
5031059	66527223	60740859	14951915	67997308	5702318	25453706
3278886	32908294	65080291	86438032	22256150	76837850	36138204
62389965	30832906	65694224	43302037	8904750	55068152	17198255
94672457	10455055	46950866	53757259	49484759	28600077	66417396
46599957	29331197	98720101	60743436	87447188	91543265	27937051
70663366	38159343	12056927	13103590	63782839	94188917	97883193
5239302	19167276	54568868	64374811	93798891	83558746	90685488
56442063	12881878	672990	7622377	84363068	34797903	72687148
2633281	22165019	97305431	46506158	74964715	52504507	99844578
18757857	79659706	60123894	11873689	92806912	8719688	83459336
68223979	74099313	723383	83083552	10247228	19495305	21544064
77118677	92065322	84531746	82368607	30320847	21924352	37830987
35882681	6679314	67614371	12655494	74567508	18485923	80116922
37464457	59552488	4324038	41080614	50993271	5994374	54190726

WS-174

8073415	22123999	17086796	46154145	47533777	58557322	60163310
7888195	46070382	29678180	62031760	91346018	81250532	68583288
82901192	46155991	22081869	39859752	65863911	6683753	79281537
24804832	2990844	12729465	8010460	79651255	90818740	46944877
68824981	69846080	72456359	86825590	53267858	48677428	87315501
10848762	69707150	56439740	81749773	69801332	21580645	32460899
95585249	53217996	45727732	54043252	41877286	48333987	89428433
32410009	6142642	64465277	99163708	57893500	42016484	71889890
96704351	70732953	57163593	43837475	28207083	16699743	29348922
60482761	1985516	21274864	20270109	74790403	78150626	20009076
14227636	9672517	44569768	93508219	7639752	36279649	14939967
50280482	86719840	31135432	84179807	69266988	21985497	13110524
12640466	54458191	37855453	53063095	83220185	95863165	86826608
81 05882	32220728	50064105	39474638	69255085	89787600	30300114
34748387	21897434	85889484	25732168	52247381	41356472	26706938
93723528	24840347	62562450	71129162	63106786	399282	92391766
6918269	6646240	4430950	95902022	64332422	85473419	15305762
5933910	98068844	2008070	2336038	84144448	58671129	40517019
33825471	96982808	43070340	56484672	3583984	25885165	61338797
58721022	72806940	17137213	98892659	83247230	51377388	74422788
27512134	62423009	91973934	63772955	84571102	47774476	52932582
69623432	94686682	7974654	62385977	51074423	7233008	2404934
9036982	16876679	40795371	79699484	23132852	77850020	19181868
62412784	72763225	7162657	36580843	32308588	98451915	32420365
29038598	54707593	74624442	9520793	71289948	22880202	83927146
54357427	21769727	24972391	19082226	77890774	91917484	86353806
68758539	79770898	11875235	2080722	15222583	81890141	13326019
12845357	47018667	23637696	52867734	45716217	28190526	87794855
74096976	69197435	78610466	71167100	56623663	21316634	52514494
2913544	89998506	57854763	26022415	30094237	89461116	71025614
20603982	71949259	48342864	70717672	29936967	94380868	51739863
70896851	37649170	21343356	68015183	12211481	65995992	76818227
17526903	79061146	8504941	82068731	50029747	12451356	49083696
64271263	9938051	50465590	59622711	7898654	71863898	62915825
46665917	5846922	66871604	13170357	63000810	31984775	62663235
4824591	29603127	9815035	4839954	59136548	95863707	55395127
88740415	99153869	49541609	41741079	39391335	41856144	6805976
68883856	29671222	509437	74841511	76233099	32329063	10699129
68052129	71512730	43898881	10828983	97181310	74332896	14541101
15510859	4289507	9190556	42200060	1695067	37745414	44491732
45710093	94699914	25939276	97901969	51838702	32829142	88544037

40 95219	75423473	73224081	75061282	84739408	18664169	79811739
42096396	38245272	13462745	75788515	7694373	96463731	70307974
47179073	80477219	60070980	94847114	92007192	59851418	92220321
15725006	53805183	55488880	41812840	85256829	79058243	51408434
8274031	28639381	98362112	56250246	56222984	23842674	47875665
80974260	6222842	38922576	8227351	57631039	40529405	22688995
73667043	54066429	22795998	50850195	7561644	6186483	89825581
25026651	69110780	56781067	20166523	78854914	54788613	72149697
49850812	78987458	73375204	31433978	63410207	80417097	26166922
68783365	52698376	57689878	85155076	35150312	54423435	42029386
32191309	27765335	71846424	52456961	22918564	89371844	11338386
66233508	47649829	36067197	78111509	79036515	27578710	56267833
77559686	38887558	96854905	68826016	98962343	30307508	49329399
40509548	92089338	89052916	31953920	60506454	80492335	84492523
52702554	41006202	99178019	57832420	68565503	18057999	75104977
97259224	34481470	29273432	84307827	91389924	9168345	88456526
31732614	8818056	32141618	64450939	98465232	4396421	64068227
69561793	35512258	36015853	28457921	93865787	289516	91500345
28486000	78468140	93833110	20555544	51485468	27884240	61397579
11170409	38383094	44107309	92141869	90441839	47296753	35567096
98387633	63036156	26726459	70949706	21842155	7194774	561088
80859470	48828522	42872379	18369692	31572084	56246343	9881347
13723103	43403748	40562308	99920411	4581949	48786393	65673467
60331099	30449858	27338548	10999874	72789637	67246111	79689029
79722025	68429994	98279769	21506812	76722512	60030869	40200152
49092881	97753337	81978452	60468960	16393739	10884773	56033567
63938232	36466049	24188936	32561116	96282960	85194359	66504263
14025392	91576530	24212435	63362319	75425716	2105654	45512348
34732783	21635574	25438586	78232574	43640506	50846311	32516027
35814351	78674830	2784705	89422954	80388715	33040678	31164830
8024484	43620735	22518797	9753747	48002671	29984490	89104196
6060946	47923024	71083996	472040	41348205	62578008	23167195
94148769	42195360	3694020	34372133	40450490	41302835	8663358
37536479	72384573	80632574	48093635	38815460	81884814	11358520
95765324	18751980	48547701	61759040	69691754	25977974	11334796
99458597	32109768	15583488	34197507	93988162	87457510	88690520
37702258	38401611	78323951	64089849	6735782	81652894	28789355
22411538	52970861	72733203	90277068	64778469	51402732	98656840
25937318	52093956	20166121	36101704	67695983	1779826	6237196
40071157	70547680	79618998	67629828	2647053	33476654	82077696
33794572	33049190	2186015	24498843	41748470	48405463	86900636
16952338	21724555	54355429	99780410	92654056	3594835	62706290

57472159	157195	35494984	86940569	66591865	62495468	98445010
63266018	50673835	12319028	4463273	65677388	32931227	39134879
75081922	61580057	659809	39472225	35724284	95077955	9277729
78601656	91762681	26456622	31242207	37283851	79080596	9146662
15348778	8085599	96444619	69526709	80609571	72069831	84789486
85263873	8782160	96069579	52972931	5951234	48550711	12217591
53875938	32678159	33452974	87008762	93877769	68927541	2238529
25409529	38929546	70087254	11975825	13271539	90845567	90995443
51836650	41788334	96103906	14165067	85472007	92573592	21710779
69466943	9905856	32113178	92391748	6411899	17070823	75002820
50719370	85334355	82407072	38663105	90853919	97064248	52410890
23296791	54246138	80580809	68253558	23410509	35812433	24718317
84645435	59402780	87757633	16268204	27849954	36399775	38839872
97546118	52220618	39339785	68064373	13303521	51028039	44551635
53015086	40974734	25203730	84580055	99203414	20570129	59132462
3548418	99096572	48294949	64966737	99334606	33698748	21716109
85190933	24434263	47254626	58424448	80294261	60247963	97628632
91774564	92309614	10942369	65691592	72167858	23878873	96878025
48141224	35395476	12486179	9076940	67219497	2081821	46139271
88591521	71268902	7485238	69933116	70191301	29867980	32710625
7251166	52115630	38643679	81239315	71070238	16089924	54000814
55985693	59340694	86006208	88145913	17827398	76765740	60929153
77482237	7602712	68024998	69035301	80915580	90674639	19677201
38305790	10669688	45134463	7403111	90063245	21738	74654989
63552069	44697999	76979726	98216148	64980740	81894806	87365270
68755785	2296322	40609490	85050844	55721737	15796793	45044605
29685751	6581059	49829783	24277812	75214204	91720950	60454108
1235593	61803881	13928169	40887264	54759461	57362079	42903705
66386939	68950383	26101500	33853530	81022528	12291058	22927270
65850571	12416767	69668472	42105439	24204366	22322643	9533421
23883294	90626704	13363633	61800737	34737792	19805243	16460413
7486499	19986529	16061830	68976487	36993200	89868551	25884285
36582941	91328798	34423309	3721022	43185025	61537973	24726822
90141868	64333146	5978750	17314943	21855734	97822255	35783016
2475811	9180733	34298275	13163017	19959665	65639437	58723506
37573668	44830841	55948562	88526063	93553826	82680557	81838085
4027154	10741696	57572706	10138351	17400437	11024282	88338415
64517993	48867046	52874405	43620916	25448199	73594975	39940287
49745313	63916077	84297888	80224868	86513510	7833453	1186467
91282934	86370396	66264732	28362025	80507227	11536092	45479967
63325781	94023821	16862347	40265685	76923854	2839367	16542705

10-96

60986944	92625934	17782821	30700539	19351144	29884554	96311480
82202772	62728436	94206119	44932998	30847510	73875719	18887489
1010647	8617070	15217523	73761698	32513606	35218234	86133466
44245319	93026618	15397439	34822749	78523478	39969860	76376358
35909886	81877864	49334406	47147013	70398207	35445803	45574959
51253326	74014421	28490091	59337010	38583832	8256936	84599663
10302276	43992150	84503416	90705209	74355505	94288577	82911572
4003530	17248135	21484365	8299783	81997103	48281533	90205647
80399020	54501919	60036880	75081798	42386250	5036388	82223080
97155160	59779013	68613812	34892806	58110975	20027405	92229507
5706927	20894773	65099735	96539006	66196777	88087859	56293368
55845947	20368645	28198303	52205101	5474089	47445322	54309275
9351589	8921994	3299486	75549461	60968883	54824424	5830545
8812686	15292362	31622926	10120903	41943280	25806189	27428168
66523229	10699435	76660415	66322580	3255708	51961451	37665853
88010560	87511996	35651325	55886509	87702983	39722464	96039468
80465719	51806937	78751239	4812630	95773594	30956407	50320733
11087505	91819459	13102954	45891107	86649520	66183422	86421408
91466441	14234489	68241016	16507395	14124566	62932925	75457671
95746299	20418169	79213571	33140652	25026549	66241351	55179023
49175155	13544183	70442104	62381480	41250343	24582635	60454870
22671914	9025995	61242288	28117757	53948423	58535201	63410076
76731850	87384514	73250428	10939610	66145808	87708879	71368970
98416140	99752256	14258766	80104548	92016551	54700114	84244650
58946865	12798624	74519347	5814600	97124370	750467 7	79342946
7511127	41724271	38736201	48863469	86465206	1635276	37591921
21447015	31324319	65528873	89325347	45478136	11816722	74151146
86097999	44443311	25724540	16373532	39039599	79052460	22940805
43959328	55298045	42017713	65424052	93327228	16337173	30202352
43907176	133685	2683820	69933299	73607221	44659201	59269626
23146471	58321836	12221239	72069219	67572913	12146684	30028739
17487746	51251783	30607278	26638544	48735356	99966021	79707150
52624821	1731966	2033501	28995016	36633388	34091355	60546812
59015641	68533088	45694521	59286977	51253142	68838196	38952708
50583991	34995512	8189540	67686882	16310307	79723695	77576983
42335419	51471277	31321118	29002267	93731610	48090533	85768796
2820598	25410598	60459139	8548218	91562782	40111636	38237938
75580649	65659953	18726209	61456445	12910686	5783319	22444460
39597088	58666106	33429447	96243971	62111114	42003937	36997443
75013432	80817030	2654668	55145680	50705502	97857757	94988911
54459536	75692528	1628650	81307095	98785575	23587178	12601593
34638217	9640824	92517345	50002356	47203581	98839162	54683128

WS-174

PX 71900-10-(174)

42292649	56239975	9917442	91029378	66960243	90812958	18505380
90603608	34723585	22344882	56875504	93048909	53024034	92182649
29691150	24246132	1505455	72355299	62412895	75885839	61224936
76848204	22163367	50831938	30832944	66763226	51325672	95961162
25072046	3969947	12026884	9440911	99997295	19972410	21524187
54816697	56892708	73804797	37000223	19001703	94346951	88222370
21205642	36206655	89907045	35042361	51138325	60071517	868319
97509825	77175978	76191650	74259676	82814837	93562108	47231462
77871447	85087913	24792206	50453760	41588539	50249457	19241633
75743890	94496580	28800028	54225228	60889545	36985498	7416277
57791421	1827594	7852779	76654736	82515322	47983212	19441541
7927412	75567644	70962557	73853002	50909338	60739338	32002806
93573225	91535607	80940298	90430284	41698171	14421279	88461330
32659353	41279995	22766060	94149333	58061613	73692802	9502067
81367501	53275820	61799774	7647009	1817465	14831140	75804684
59901907	36164407	83771881	46727866	3744310	57557049	24136271
2929705	7763259	21074949	33475016	35998046	11827675	6316523
39542967	12547852	9511412	75822355	20145458	57812492	44828808
14193116	54390026	30626077	18614601	76163278	81414748	40234811
13693528	16716440	96409970	34317116	81721307	74312028	79168195
57532343	82619941	69668306	37435584	52613601	88747989	43787562
75431366	26831773	84424441	16491458	75527339	42356470	80074256
45158847	22276729	18667061	75554854	11092570	87650790	47842039
74543349	9941927	29687599	58568356	92284593	78972227	47555162
57282439	31728861	37457635	3262653	30101764	32772622	6498874
19531150	90399116	53387639	4117884	12989858	85539005	98289974
40154888	47028697	10901140	66777719	56384553	5643175	45614860
99992640	23237287	62989321	51357710	57719007	99295666	11993953
14534699	57928089	75538178	78840015	86309268	23790472	55742055
50504833	89851397	80184799	67908564	59323621	8367658	75866004
40104485	43578984	4184945	20478535	32578550	49859309	49586669
30182937	34602020	65101157	2344490	53477734	10638118	96542233
51778616	87341026	57820598	4428542	28160999	62457866	99243975
34235619	92399947	19661044	86418338	5102262	56319484	54192754
86866299	85297343	10824533	24073904	8570123	36698029	84412687
88485178	32454405	47116014	51353694	78960011	1112720	29923745
53016676	19407665	17267573	31436550	1547094	35769700	30527886
74911622	35305594	82074045	65303630	61100355	61839508	51688336
97752108	75697782	5361816	21250237	17609140	14388788	68586248
57148417	19564673	93932467	59775268	97478495	28921951	64869931
74243266	23827574	25740869	28585449	57560740	76168160	84535757

39284457	86767075	17756843	27645078	44261274	4996251	26844696
76769848	99580839	9398772	38544356	80087526	15812715	40281283
62423230	98191030	75268774	16252589	66633609	28863693	76101763
95726498	254194	37358148	21694467	13236582	46971424	10329639
8452239	75869294	88793295	39919259	5448920	10488355	44210858
63105776	8712931	27591476	12743348	26713620	13263786	6958579
66891401	33221805	26293314	520759	20079807	89989447	71445562
99900606	16214423	66702715	93881602	42538444	8210646	96312862
23399609	65972958	99922342	58904174	19789722	19289485	2849280
26976339	22008831	44423123	34781215	5879285	37772236	3224305
77768952	20306294	18992490	55470575	63733820	57380635	96572131
32771556	44043527	53428411	59469739	92376881	2850038	48300132
76555373	25713930	76053182	71239881	90807674	4075447	27126397
82639043	41171168	48290262	67331401	63118720	67686989	67808
28503022	20454248	20433702	58236016	13881771	77812048	24708515
35145665	57913365	63980884	28377816	93959018	72894945	76944345
42414183	13691974	67027096	92450251	41585335	22550791	78074005
12017240	42841524	61963007	36258582	93379046	5110930	31729207
47191237	20017477	81373899	99044416	57455395	65429054	99824035
64494003	5890602	87702712	32098786	11029712	80854471	10891158
19209978	47887218	3326079	18348176	97380050	78073532	64002198
91104984	46398450	22493103	65827868	44832460	33357177	76435235
81604560	11251256	31041573	30262099	11662303	51749069	61441405
11556323	23146363	55283617	38787388	97479382	53874729	34759726
72446806	86519352	28220901	2769758	71592302	72881541	10149182
32171964	20429395	88642636	32823403	92875131	48784820	21422375
9248998	75657464	79109330	12175249	71091326	72459338	36412060
86605103	24619285	17738176	42319073	92498350	52512495	63041238
35473799	27842384	91878608	66449438	82320181	10194636	47715566
6309563	62608664	80264821	37358428	29571330	8954514	4180676
34602231	71036936	18773669	1909775	38038884	91652394	10908608
55462908	92716321	10381826	163874	46643964	59361355	46438777
58473909	71304434	27742162	91040239	32586084	73902106	90108811
73909146	19720201	40007372	57909150	79623689	62029799	36179844
25832498	56123968	56931080	79530811	57972970	27182226	42588435
55690336	71958812	48649101	95736497	13107149	29685669	4274263
68302401	44874202	67805672	81280890	32684214	69572413	24161419
22112037	83229131	79093489	13308268	50351274	64949991	44762133
2127050	25875218	49948688	13828966	68715051	64341068	94619987
10782162	15425200	10479236	56113759	38170478	56867442	34119059
34599794	2480095	30417105	45466571	89127324	84100324	25349567
26466555	42238473	8574479	25021787	66496142	54003683	2477682

10-98

WS-174

12981277	59387604	63484831	32230019	42999849	86722149	96091422
65622679	43888750	16352384	55459761	27953567	55380917	91642791
9195788	55167147	40419344	67078508	62359854	60149771	29666912
13479735	42163433	45058789	26051951	55684411	39496489	47222104
83050883	30738310	88906814	82612532	47230493	50611874	14084060
43532906	50111858	40668246	58370911	50604893	86132438	15325979
37800653	90670878	48091778	21079822	36342839	47748738	12716355
41857656	32980525	83791893	72827620	62028912	11227067	21653627
72299373	99734732	15615903	6560985	21985866	23491105	89769967
71125565	96134953	82262170	20898410	33196834	7133532	22552779
55076368	23269439	27954174	15490770	36555462	46771733	29836037
73625288	87388755	58339046	93713825	20314361	92464982	53339550
92883674	20678087	9161852	40015423	15961643	69122726	29257330
33985113	24573612	75058409	3917165	3576915	95457957	96162113
41011533	72158111	18115095	93523576	37004383	1811657	96389516
92620092	35736102	24884661	42429265	7159946	94533656	79297487
70851534	37600091	63207285	9030120	92272723	13484947	54567887
36777658	66113280	20880496	66117291	82022890	89896842	16450548
87525237	5489376	74840737	54459199	66212160	1643679	68510971
94595363	21942400	8680171	45522577	54056018	54501573	50303327
58298842	36450620	41432292	12165319	58052558	87396129	62970350
54543009	8007003	32892766	30905148	84715273	10795333	90075239
68997344	86864202	65745469	8513653	58716247	39395326	19911848
62645403	40051844	82615639	417815	76884804	34820520	37852133
93883731	25652094	81334670	69209574	17447433	61990482	3869416
2495834	16067526	94996662	4072533	79368484	54848675	16962653
43467113	78294119	30172706	15222521	80145978	49551436	85022628
67265347	83958619	65485335	72488935	63718264	22417805	95054210
12380238	8807319	3073775	34158854	80357153	84673501	43639422
93793131	67601051	19374005	98835325	80959699	963355	93670773
17149933	77471938	43446777	43079709	51614685	39093558	29914687
4478523	92031932	84739905	32645602	57094801	99062061	16967523
46250542	43529083	76782271	46407196	3781781	87315142	749498
12435336	55174249	49092365	83237404	43391209	90472679	92229307
80596	11972649	58143133	85283885	34881914	20136453	72922401
44020678	81736882	97944161	66628357	15333972	66975678	93650574
24998605	85436772	40878848	12126486	90865139	10048225	10656829
86042704	41096944	86530724	79698618	81041002	97617087	35430071
5660957	69859905	58723296	31666021	14494993	14674447	96956401
48614550	63283806	14208790	16374272	116771	63735931	48090170
75557005	71603759	71623062	64516594	66400803	56314454	46908261

10-99

WS-174

10-100

39184878	5241497	80916258	98152961	78066650	73544379	92986114
83912465	76575113	42593412	61483852	9522924	29152739	44457275
15901777	87496341	30290464	17427551	8823860	61200012	46783492
92200704	737256	70700808	92395661	26266143	41461948	41131291
226108	52558873	25553821	1107168	7954528	33079214	41504436
28429466	98685932	39037036	6951030	37029850	47151857	83070443
44122556	46972847	50360984	69672898	32398413	82581163	61569027
93020846	5240396	49943306	61868743	68810882	75871071	48429849
51932412	16281311	69324615	32515603	23886530	73657943	66503830
1844386	77586333	73833175	30013940	71233073	33505100	77004366
61261956	44540222	67636730	62594003	70479821	49681261	6857820
50946334	27756197	83214823	79238963	35087334	27578574	52441928
91140709	33388584	20442006	78699174	16533057	37816474	30261166
87829556	92253040	69003063	55721899	20354121	22267840	37040242
71060552	12042177	814873	64946731	87270668	91799585	34002252
71856268	54353688	50803200	27691831	17323227	42551457	89179848
62656509	84046541	73940186	87620382	24908370	97113778	13642015
43965540	7008735	1467646	82467406	10936818	43750507	99914666
46123968	60746000	25359021	23987759	29141324	86986616	19792871
71068770	35788113	11645125	24098960	9867506	27006060	52803360
58273457	69213923	5574270	47130293	50392402	13989117	10624866
46490629	40761440	64688418	68255963	34338048	21107149	91039620
86649190	56899976	9830204	51380915	39425405	60777510	17678213
40694462	59534669	39917002	65788383	7795914	190024	35585767
90679709	28086961	14246717	9581701	39902284	54399879	8055163
45535956	27776237	16319095	58509217	49067360	50894028	32696895
23656757	51651652	5294598	98340224	83924961	58154210	57712337
45875260	43500379	74597635	60701904	38032144	57617288	56344211
78715530	74001211	59524122	11647901	49250168	93311823	15180582
95613365	38861220	27012034	86644135	48912191	76110698	78647373
35680241	64035057	75958916	692287	13607998	26535152	92937371
54868520	38364421	9878023	54431212	84197919	86503808	43077151
23152785	52379528	80555945	39876746	7062332	32844913	26898374
12690628	91423014	520690	18138723	73928976	40699993	80913121
21498901	14568326	64387132	48299231	51208752	28033614	80390024
69865013	68202058	21545502	88983253	85484407	38752921	67389750
93959880	97144431	26389950	34696743	45461287	74697391	48430922
17807195	45292841	33653231	83424880	82909838	89440969	76890797
12415367	30284156	63487648	32088719	94528937	71912389	84870594
85264714	32440881	69316706	78457806	71556051	92753985	29657123
6534427	80171760	47853394	56359884	71116255	31002841	14408688
2691127	7055840	85038974	49319359	26503195	61376550	47033542

WS-174

81455110	59742852	31354001	95226267	20790073	72505546	25489798
18846399	20355808	69725941	48190744	86297717	6190190	59891895
22945740	48571313	54915021	4337197	8611869	3122517	63174668
30984047	22573290	39919455	577603	8582828	23260240	37606542
30106748	38763059	84153005	30958239	17630293	34802868	78143083
76247464	33787093	91088004	23546876	48087723	6718847	33720053
84190092	535078	20243781	5228849	25653184	66965777	92556867
88079773	36247975	63422045	10786436	41623063	10160021	68655508
98105117	74313062	29290161	18051359	68925952	91755059	80472609
72647302	89658936	34647601	5192812	53503948	36966861	19997548
57609073	93678127	89812390	56595184	25461620	91695070	3495535
87805306	99665439	79555102	70230344	10802398	20389921	89856968
29700758	26099611	54261774	15247159	67292447	41020082	92958437
58483857	6737247	14471580	27116665	77297240	37822454	64270731
29 14410	8138241	16555253	28347101	63323602	59994280	84632877
93367877	86118965	38961384	26218767	50873974	5415612	13003310
61312835	75257610	168326	37668361	24347021	8911946	47638846
42903617	63911353	85621681	8183521	32580183	95798300	6897316
88292328	24997236	46924537	33498044	12725141	20586914	90810527
84335057	52108503	94375933	97515197	33816876	62372835	22673592
43008356	57602676	47937662	80221968	39792198	44506328	53666535
47405387	4613485	86945928	83131672	21765153	12951448	78743987
69237329	8076809	29464817	99575364	95733173	53815258	70479841
50243894	13558935	33537113	38058919	5535964	43260967	50428416
87598419	50954180	60578510	11208723	42477473	21153456	94523152
1193296	14958545	36039444	60648789	35293898	67203309	83557514
56027290	21573566	1124198	84383417	70378025	4563668	71382533
77312583	66793242	90419039	76983675	75090322	96376281	26113910
53863115	74599140	37151860	72768368	5784867	65992261	51797706
87503395	62126422	69990117	97923609	68466418	49205321	39651074
67323257	30231132	48229819	2919556	56545263	97496313	71477127
54043729	55296085	89982419	7954024	18900861	76066957	56100235
60371169	83943422	40621768	27798111	54365430	57140107	54226082
84913977	97740720	23769959	15548342	7426414	74526545	74088520
99751623	88134492	99188958	97129131	34828832	3246987	75060750
69773368	40213432	83355530	11027804	53200373	86834500	35485441
86915648	36611548	90784665	27882853	22378974	76415766	70780294
46766959	29753248	26335137	65522291	38380527	60313263	65564584
20193030	53914270	37278908	74243557	32013886	2618771	80601587
10998713	40128786	53510347	82763589	1746500	8245781	39226515
30473232	13799196	36542070	15301780	28131390	34814082	71618325

WS-174

75304142	71688842	71964781	82349218	21747113	17860647	72591018
45610841	72241818	91230105	76598239	98287029	91524894	48001342
15562331	85888676	3001792	22423903	22212780	98724092	38799140
25429400	46138921	78745442	11457975	75026714	51808478	34505647
38323798	75045383	40127420	79614785	4381404	44268358	70063162
5316649	47582693	26468646	34636721	61291385	67120308	30303451
69417006	81490444	22204866	10307908	68211899	2704385	72949997
15035442	51795616	28708237	22129380	34245500	16219756	82511099
12324691	29555103	89333072	62456190	85886259	69225311	57502093
36943766	41385632	41719609	41804156	4332028	37684044	62882572
1121762	15854707	13626158	535082	20356308	83791768	69311163
56741774	96628006	29010711	75506518	31252749	65412295	31018825
61411867	89362420	14661132	42633013	76226083	19159395	41968666
68867725	64345960	98022312	26572804	78408594	29318360	6029511
61004256	42770222	69385712	6444822	71438226	33436070	14124462
60007233	46369012	50853072	88496414	80106217	4750554	94303486
99673222	30068311	31208486	28180432	72269392	61625973	84393296
79855181	56232170	13824595	79969318	45869324	10728607	92010667
23391188	97511753	25000414	2109206	11218256	42221077	56067133
68681244	52818327	76666743	17638911	8806009	95892763	21314607
64564621	75302484	25046559	92284842	85977009	72036107	75601672
12584410	88122914	41916121	67084957	17147527	9787178	48947909
45982990	20473228	17501589	38025089	30551227	60443925	29613229
40190935	22719157	16574558	18499641	63373772	26862689	46071792
69343813	35716191	1626205	15746898	67788384	33904607	41508027
53919836	59641968	65579699	42749801	31780112	2838091	34999879
82316092	29382029	52313344	97738729	27830081	14210322	59471967
20836479	35805816	37953338	75205041	80142172	76700088	16164827
13662468	82471354	10456873	98094214	36029091	79868514	28658333
94642312	69896537	15500232	32195493	11250449	8339328	2460122
85171712	71849014	46857467	25756183	56740419	84630591	44469688
96664545	82393424	57375471	98736092	88360062	2395274	5720658
4517334	36234701	92655650	48436692	25899972	75231018	27855673
63066978	88493940	10508504	74164649	93902685	98767200	43538942
85696795	29095534	45798726	37955997	62959857	27793016	45252379
55841576	31622912	9727060	26368304	96847594	11901766	77353002
78708944	22943864	77445339	44749339	44863849	44994111	85601007
97678340	13747828	67870260	55510559	59521814	80937770	43827407
13413044	45083468	49224408	39729048	12822823	84188610	93061694
62619249	61948410	28274009	20390646	79621814	67901074	86807878
91870909	26079690	30722001	32760848	77929960	53681884	76544964
79082457	61499671	20970023	34523073	926047	12994834	91304389

25514543	66553069	44841632	91081787	64917105	59148645	52059252
63250009	2967366	92344469	18552696	79503683	89669042	50510643
71391298	55451077	52093737	93846012	44072031	49986717	9905823
31184834	33899185	23195925	29432474	76204365	29300476	39793917
92864642	22147355	3041074	53749723	73174477	21795982	50449180
635448	25245635	55477477	18501867	21352256	49951450	91528595
17898866	74012890	98238772	76152933	58828241	31152465	62829763
64742580	34861130	3878182	75279592	52145369	69944578	24795756
16103405	30590462	90447557	73935824	99127834	20269577	59824363
2207054	79489806	63175418	52082788	54267985	55755135	15813687
67625250	8078067	18774726	17341670	58157914	98444683	97407697
50523010	64329565	39457024	10748191	91478493	84841731	78607144
69576364	1091857	52079464	94976094	96549307	61783318	81583417
87552329	96543514	33034373	13926195	24582059	44251037	78986309
41051934	49483794	88929017	36131012	44572387	32967294	48940032
92825406	14036572	3435241	2248078	25603240	49168111	66614042
15282525	23327299	45038955	4959632	70433471	87956364	54209578
23282339	35890672	78227170	25834774	58595903	71768841	73159723
9394739	14843325	15935708	10616911	36825152	60020789	25068533
39361173	5273968	54855108	63715857	88922641	48739264	82034012
12850937	69775570	41103482	23219301	93884299	39996068	8344386
16180354	47809481	98157045	58738598	59483411	74339674	51589542
2868014	17123838	14465059	2113522	68501023	83177122	92389288
71425758	85343788	32577281	14160240	15507749	51017790	24665766
79169322	89236717	18885973	59896072	6233932	82467166	83729834
71835750	76372388	58444946	66051556	29312730	81191759	75418567
69427913	83483517	3306726	10787182	38073460	11945165	90279115
33004191	78466984	61312918	1790580	40329550	88429809	85445574
20058214	19413548	3285779	92602059	17228214	43185494	74731719
74610255	13070197	63906604	86242183	15626635	40034443	14154795
45096024	15532655	80576132	70899539	45882875	45426995	49530797
69023884	28171939	49163431	69175628	2014393	28326278	14409077
13634341	96518681	5529642	24476839	37028388	6023378	64814859
1371808	70740481	503230	63830951	91664008	69194185	21688546
77679726	60368139	16517571	4822860	76959997	80077652	6476217
53240899	18938370	57515354	7344450	50445008	17487915	56006033
94667631	8909420	52225249	35399748	92148900	19800045	4449802
89369057	67153193	15888807	33391473	5328657	56293559	82169978
79703061	71812214	85354413	5685308	83805179	43931595	20912193
18282241	30437567	84225569	59037681	17466109	68540966	11411533
37296746	39185094	11317934	27755826	72777980	7750872	75261331

10-103

WS-174

10-104

35899603	61035519	10753752	99039266	46795104	16212290	6697896
27891666	1867787	64806536	35667505	8563630	45548697	61536584
46256428	28968544	97238603	78187601	67682656	12391078	18083045
20781454	98474274	40435305	16776848	50956479	44716373	56999830
316124	34721034	57718662	89590246	58812349	11790960	47204660
71279361	40751015	39851378	98724894	61360727	4559560	90035425
22705484	658979	16122952	43608997	43193917	43249187	3670756
51005968	94746533	81069559	38511135	94424873	24829414	23434529
63960666	65595846	94338439	9217110	64934245	4454288	81349930
95845434	89914221	10708681	68330235	58282043	49852583	24298517
35390804	95110814	89567766	20021511	60639265	9325398	77432204
76486661	51908006	50973819	75188360	35775116	48671198	46273022
93132031	28049497	12729893	20050808	79505973	30327580	77117052
69109594	23416925	16229771	95814161	50440301	19289946	15817973
53979793	1501100	84059673	89729240	52742502	91067704	71392202
80882092	88125896	41671261	23872214	73578365	26746843	8357640
98997489	79498534	89491660	26517477	88875244	36825106	58726733
99552898	598340	55078031	26675412	12151362	71138342	52919916
15219807	3796669	50615801	20934295	85603400	30779661	44219299
32129614	37013388	50798159	20097889	61795108	10602438	32328904
6226196	40395076	93028079	56497786	20966106	58549111	97609753
97547975	43731385	18550088	6136328	21059740	8273870	24110185
57209869	3480629	71128051	31852518	26472732	81789973	68391563
90893986	16255688	19595877	32228512	605446	40312591	54289951
2606414	39858114	19784261	97290683	#1593620	94359583	67161897
15883705	24081501	55587865	7852353	99479320	78211128	64264910
99653107	1070997	64146284	17222883	33129426	57245832	75655216
15869371	23495552	80653706	79971788	58061607	73524013	56927754
2607593	73025334	31937184	19143558	58430932	67073756	62895960
90177990	6762939	66141402	6692510	98123042	41702053	84795745
59766944	80491631	83729572	64465257	98601075	67788352	33004394
84177709	54833986	6389967	42804820	22368257	43364283	4088395
37451404	95003188	53441981	38563104	96232646	11955585	72096739
36449672	80844646	39511672	71691827	54112147	36606241	75707704
16048634	62905518	45607205	18862554	72169500	70071050	93001411
55541019	32174156	82093981	89265378	66129109	63523268	39784591
98944290	59317665	93293342	2592032	37921846	41759827	99461205
94619437	79409561	2453086	21455120	62440144	37138919	28435515
418853	28332696	60740306	94119081	12329175	21479713	11649056
81742228	8776997	85043576	1416100	92520716	10616435	23434485
5187451	36907877	94495673	3284618	59941209	68052860	92076969
9527940	3911425	23234053	47639114	3071977	91121452	71426796

WS-174

PX 71900-10-(174)

90854196	89876961	83945411	96575629	18005342	68381676	81191776
75896805	33803320	10709446	89850951	67638081	59914007	7673737
52809825	64923099	93552053	32803134	59545542	77357024	43427433
35771900	92417526	60501044	69640650	64732921	8566985	82638626
72940929	28373056	3434770	91379517	90472587	30337686	92980095
48805766	73797094	88737545	18416021	76290866	46797601	86457031
19858538	39341921	84204176	72421905	57101669	46643378	42876206
98100945	91165528	4770864	37404645	21253881	61133528	55526780
34260714	69027151	2525465	44330206	65119253	19836548	91815483
35469224	33357794	93792466	85132139	60983197	39653436	33215827
19137345	17866692	58995659	43277284	55267665	44312117	93116670
56481338	44010776	71612697	65300969	2847143	52804953	29546565
71276482	46712529	85109401	92414282	17412779	89789703	89460983
42858498	2598795	93958496	58210222	93560248	11944045	58771662
50104332	97384871	79478509	63026195	14942507	38342638	68174838
33858969	99812868	66227195	4272423	65460286	38907766	83845038
91489380	61206867	36296861	96488817	4387310	76196181	67506474
22155896	22903051	55618675	69282254	48790148	37090101	46737032
395525	33722245	11636950	9615878	61833736	23530156	38127997
69766188	52275408	53246873	52779145	48154088	90722998	37918592
84437152	41789871	5124531	11694653	22283989	88685143	20656080
61157176	49695501	4799394	28667326	79194823	1315666	30470762
78531738	3901868	53870037	890995	38792791	71917629	98062734
89968710	24950359	70374258	32809455	70815782	8107414	39048017
47429270	5393145	52058072	30054657	49752063	13240186	14140178
99471128	70871557	64009293	22263615	52401730	34046294	865889
65481239	91479407	7197025	63885441	27762895	3205188	46571332
29368960	87313938	78370206	23142238	73458272	86572861	57122583
98117884	30816708	12670871	91824424	18558889	88706919	96380072
44366227	4707748	98059444	78366244	45902352	336789	79190758
21178219	80645278	2824987	14662701	34957446	23733190	54917370
70418453	68128364	68645747	42911648	21401213	85015569	2901728
26030335	4869816	93897641	91088864	833951	87256213	87809706
89226990	97105302	33032761	76881979	32785895	83605804	72327596
88388044	60562753	94197158	61280700	34969038	55159598	39588682
90626876	18202277	19531659	83481367	61369858	35824537	96788405
77119197	63987892	91306721	84449417	48019016	87143196	75194705
80052735	68989190	33251105	45244417	13583283	2349956	73027628
51440248	97948113	86715835	52685884	43278848	16579615	5392755
41086727	88744346	75521674	17208377	35911185	91333126	21959380
83702477	1538362	58559551	57521076	34096048	58350937	25574142

10-105

WS-174

38152134	77691288	18739242	92859873	22205223	20350909	66126695
33114642	98628588	36468188	84362545	20085048	62271021	43186962
46451600	16552928	59949665	37498673	82581075	78268456	99679657
76877779	48850675	50036751	75267734	51987387	54834593	23465881
51265759	21122094	17564578	65179398	55909606	66371162	27770457
81719030	10256251	91410041	38221923	27705389	63664909	91889919
69936791	74170020	25262054	73927591	1737652	27772360	1035840
14645810	96656585	9456998	21461526	36889571	51401522	82263493
26080153	43746957	31922832	27878879	44801508	36069114	24916477
56741644	42551537	91430380	73520727	98704866	34810867	34732730
88106777	38286979	16016597	4456920	21174716	21513482	22102794
91643707	34964384	58452607	67147147	58812637	21061980	71288775
915996	59904823	81984473	95631923	24057549	52867122	35984198
24913710	58852315	37305270	10543859	44089734	11130481	71187823
64669366	69162598	38112121	25801228	32348773	28303124	34137393
13494632	58587503	76502491	94578909	63149619	40952132	60460719
52996232	19502563	91505883	50061374	96864818	79259649	80147823
72623473	90398386	32851531	78855935	73227931	49150423	95998881
12428090	93449622	56224701	72144725	44197955	81124658	66804624
7960344	62475188	83460516	11667646	67838770	64823267	11754825
66725134	53476520	70678215	93908680	10348023	60854150	30591312
56156260	25834365	36759634	95939255	79880535	98393493	29519591
29845861	81555204	99180335	50468754	14413538	4911922	70449525
2711031	30117376	69310825	47233275	87618884	89797573	42421646
61260328	94252986	21201960	12503495	93551482	16739960	86977839
41316797	40007381	58162335	88596999	91156376	15745419	26181667
80931423	66945929	90789643	33704500	49311657	78268650	4557289
26148438	85668779	11830670	90570828	52092053	46472306	62176050
66664176	83457602	63909734	40076533	61698785	33879112	46302655
13079564	58980353	15347367	90295501	91316295	85346419	49236320
92424783	24581414	26106120	76806219	49015414	65962122	63523192
37646586	82868889	11219682	82336816	75511521	23147	51143169
62123522	9545072	83209742	70519756	99377051	19787262	47496049
4948695	31193024	45184039	59880724	79508980	95243606	2556995
48914855	55643385	93326791	4043640	71866965	14977881	43122199
38969125	13083802	43984576	5651994	86151344	10312123	52569094
44652003	51013112	27283615	76256445	18002024	23228068	72079056
75868541	98874871	47350276	64862621	2818601	94317102	80060643
64855821	79958808	18641094	16148827	26213814	45755618	33216195
2692461	44583467	76230568	95345574	21650153	80508500	47347687
15612710	50954341	50071671	85251721	69579481	88778223	89013630
515573	8406776	4879447	96398147	80954182	61119076	51621296

10-106

WS-174

58055116	25139172	13287181	22232588	46173338	94834930	17686572
7411504	57736763	61930408	58718880	41654363	85374432	31981984
44686832	31268998	65357821	61441047	27576329	14313385	6014638
87379656	70804579	58855543	51359247	50397633	26928072	6355553
14206165	76746408	68957163	30248025	42369059	58296031	91590267
8003786	76610962	59039839	78174219	12630556	44108953	14921833
77011975	65909771	67197966	67467813	8299834	83431817	9623668
12543867	21323077	34404277	5190289	70797863	4141081	41772657
57737145	72676700	26077336	98717815	44316237	74801358	20404608
71028694	55348030	73090923	87202843	62818324	72767429	67978436
11672650	74391840	42710412	43062775	95849800	43757847	69841091
66325264	78761068	12883204	37975564	76541024	13201798	7964057
32710292	702385	29245987	33199065	57668135	10359429	41187853
73940586	98873044	68315743	447501	92030140	34327981	6341241
14239113	64104058	3805274	77123456	69793995	83614153	58763322
91551630	73187421	62414066	5466930	92286893	9457304	30069812
73434100	77663154	45149589	30269329	49032794	18019361	30138492
42629647	7092521	76805057	16326431	10272119	932220	52433808
8872186	78513057	15245346	51162348	64789608	96325774	81663454
57424168	13703885	39640528	72745617	36850240	94699092	88653027
56703967	13340299	11912196	30046213	80643830	96308079	65005376
28545320	2407409	78662827	67773642	21840944	29709310	4546941
3477536	79181016	15555099	40874341	19554848	85974419	40633347
85099080	70502934	28799547	40693902	43780942	48453007	82214728
98363566	97153671	17890980	15300839	1659504	50922350	24135240
90959828	45594751	71164116	28144182	37952454	97507134	29278031
79467577	23926406	95211159	85054996	38306638	20731594	38007271
28987291	87751814	71241995	16060263	24894184	82304434	15420959
25390618	93750714	54806237	31071060	54473821	74900025	47312071
63834593	73719545	54639534	39060374	66210470	70641417	32241317
92395823	53511716	9127304	7647126	38137773	76345784	35150397
56814625	24684482	53707064	72713498	72813732	73989681	3601604
75245259	25825378	5293505	67592326	21395577	60683294	6165971
49660000	79839358	13758188	90878743	31819681	42237696	54253536
51 34047	30687161	27912740	57843358	96245716	51311354	95825320
95925975	85322300	41817590	80146619	67583827	50740103	69164206
83347821	62596337	1921374	95902487	35652936	82749952	20768315
28303285	38666589	54646875	18231815	45178096	26912031	91966598
76966855	4569081	89441450	90422123	29523969	18788515	2595867
11589012	3217914	62495533	70118424	83533121	40900487	83996441
55738939	97064326	54605159	11558853	94319448	11839776	78302216

10-107

WS-174

10-108

45362245	80973812	66755616	9114573	52156219	26345562	35053360
11357847	76832720	61454775	36395099	41513984	28595046	59104771
25770603	59745690	70706792	11102296	1299917	61593642	56623260
9979578	92769076	18072240	2082316	60064440	45083823	59211145
65457182	85804839	60447395	80823254	43528356	56330559	97315369
3320464	30415152	91084701	12675018	86939014	22847143	93446883
26987068	55397894	32362974	25147562	51295445	50932016	27620354
19822893	6536617	7562355	6457650	67791386	6227610	38836575
4796054	32116477	70101029	70653274	97362886	34998082	66908997
98430589	91096486	75771108	54877579	24773621	64497877	76731542
95711514	69101405	61482264	68153229	97050691	73683236	31164741
5520767	18397401	5765724	64338915	34052536	42245947	84612736
79049912	85479078	23057063	93263721	16438275	55796576	60447804
92329101	86440348	87409061	92709506	87100058	29783653	76371718
39596738	48820027	72329432	5820033	2126433	93189081	22347639
216747	57653913	12924226	70955420	7295113	4690860	25625251
89097395	75262220	48054375	8108536	70611733	2487659	54526708
86303818	4690684	20674079	30627910	70179923	68771265	35121146
39241974	833694	99306725	54666462	85406199	85738210	65111592
27462847	18075443	75360074	34535794	81102412	93114945	84262964
37279070	78800885	59259459	82432647	87041524	88050737	24723629
317496	73317664	23351326	49869573	69894512	92751360	56562765
52985615	85134447	80429494	9496634	62745951	50061132	90056958
91593484	98503318	2076783	38629725	91343395	42407882	76709422
10310233	32026227	81356061	78151347	40291999	11873838	96998529
82674893	56718123	49458525	49158885	75506855	2978576	40408056
55523768	83745899	69654546	52997205	8527136	35362158	85472943
15073256	41824168	30431782	55701677	21465631	18152284	89540874
68811432	91343480	44799073	14981866	21009114	60475747	73105549
96003235	695585	32732624	89248517	31606784	92890652	82763043
44011857	98510480	69337963	5363741	30181277	34471094	5814843
83763673	84258704	51655950	12560807	95220117	68838051	34873618
55191705	3287611	9921522	70170490	71840826	84950937	63744839
98927877	246007	75479750	45776454	82496432	2370901	10645869
46155229	645549	40968019	24820447	39613441	96885235	5219766
6458190	82982479	67671795	29882299	32874599	56708296	79415085
23634616	56614446	30462633	26140770	4172982	86592739	79452695
7924311	92430596	25528582	96398713	47363434	32365744	87185958
92635929	30523249	78682869	94717346	65297333	18173055	2774826
79945797	55273324	17663727	35832502	52421576	55476193	87559726
36197905	19221124	93007181	5473386	27668769	39639656	48214815
92366127	90957153	48417431	20927090	34706311	46189499	25193245

WS-174

10-109

25973743	3047639	59231653	86646941	45385716	96243792	57075548
81445464	99020336	51133800	66993509	87116474	49034578	68206233
83282825	40145605	54318006	18334154	56187622	14474347	91137305
16117552	25915791	6484466	16863459	71548085	36943735	40513551
70482622	84404461	62419427	71141004	30032274	91169885	27229169
20995403	77415819	18202689	31121901	8320249	96726677	85438905
59140562	93106546	58866679	38735169	38131760	41407875	56915167
34524980	45322158	53082572	41851239	37696835	20061378	49337587
2416077	54072550	60772126	89237982	3675262	43549474	64870271
83808028	72240741	60932314	32188668	87687364	2987340	18518419
54759064	46193812	12307342	78369945	15799877	81780289	23723577
52923233	75056052	71828584	8998682	13176143	91552834	28693395
7252551	15310942	17438151	93108600	39985346	85263818	7234919
56164525	26426354	27100913	36820605	66323885	39967517	10463892
27114839	25928839	82001115	26928002	4386337	48824081	52157430
6587277	90534681	65794115	34828828	3134461	67916325	9217934
7082123	52727571	73687157	83035611	3764869	14205789	66168906
94136527	40915432	5730088	1363378	50825721	55965508	28374164
19344194	65508947	65644916	92869939	36942999	19808653	65652609
80562795	98360850	20748098	99638909	4310953	75631102	8591299
95543029	73459104	21424532	69928976	58253884	2483197	63220995
24342722	22191692	42354678	56387895	99659164	37246699	55789111
32090677	15392988	43826483	83055489	26094981	58229943	76313285
6392518	35915228	97939433	92434100	54369114	82108915	49861743
22083866	96038666	57904133	72704908	99598278	69253260	38447331
60572105	99460934	97853444	34144707	85031824	57528584	41839778
66913721	43573056	14372057	45942322	51017187	7702379	53250299
14940465	64290046	1464002	55965684	33325336	84454550	58201070
26547639	41564154	97782862	1334583	97281474	81171090	30836316
27405437	13649906	29238967	54274457	55414311	91547159	3263985
16371698	61923620	1978939	53333866	84117043	93029511	96782315
40015151	8309833	96284772	1951189	78909536	28784086	8403579
49160276	70445322	18691416	89612655	74863789	31861454	9421739
46483582	61775412	93473711	63661686	41397804	79987366	16718509
20396635	10297439	42137067	15992973	10963359	20929099	91222805
5454891	44244735	76597732	89511236	40351829	84377978	51587685
84845260	43666024	36845031	82379033	55186018	5416011	24227840
11596456	78412220	11035189	37222864	3271592	96151464	9873002
47399904	72012976	95977017	68446241	18463844	89995669	400778
258993	38144229	51945391	81377192	57464226	45424237	17785772
20334062	49576936	24813002	98608731	14728841	16536869	10836612

WS-174

22426403	68494718	40024769	10444628	22057563	27180403	11721370
31045497	6432982	44928378	57498244	62882757	6326118	16958749
67858863	3329551	98020133	99491657	22618163	94311626	60229432
78599447	29619857	31922022	20085658	28983820	10267623	8670030
28674927	24588072	79188946	57221539	63341579	28846118	8635894
7896872	21733290	31649973	99909117	89516230	46623583	22881021
67709695	42747885	45130032	16969478	37246619	95368519	18229174
92270674	90060915	68692712	26900808	44680498	83429489	78350904
17015959	66658069	45875317	45103883	86661218	85745993	15624802
80961281	34142092	11467548	75895584	66932451	91712225	83440881
85392352	44416373	73220405	58057306	86747495	3809963	74815055
3070693	81674823	22757416	91076467	49474432	93995926	37436218
70449070	7435882	72441592	74058635	64343502	28874706	7557259
25958725	72252987	2779368	73502043	17631574	70839517	64144512
68101393	50101913	29334400	54608062	93225046	94578490	51362456
57014110	43661711	49730934	62064204	64525314	43273373	3381621
86415775	67218560	20283977	62267001	59008590	4394522	10646216
55916913	3493704	36296247	79215981	59654295	49751079	91905468
16759304	94285648	513956	97135619	46929268	32371173	11837727
20660456	50043567	71192588	21249948	583092	28779751	20670586
66581769	46913742	45589420	55411995	26394248	21388865	6081358
85275439	65717418	24698987	64860106	66285218	25942664	68268051
67338382	3135482	25248382	13169828	48119165	85927954	34215085
51615470	61176906	67862128	95179402	44577375	39070258	2016753
94716982	55057411	47130731	62714067	92635563	20227064	71822896
22577301	18537792	62882691	4469429	67585750	79840382	42565002
67569921	62194503	77309142	14616455	76158241	73933334	29080015
11876767	45178416	35914160	67894826	75504138	60762599	89663167
19454902	92900387	88189241	28756163	28189764	66358534	75177111
91346802	4034701	88833563	35206643	28494917	60882646	92365146
63360000	10188410	61978674	74344026	39800773	17299808	99685961
10456145	77614369	94696886	26594598	20421228	31050573	15011531
50228270	94304914	5627496	68070717	57553008	84899749	50111120
19907085	62872062	96816144	52156788	19816629	68331973	25141443
77174168	65264424	31912540	48824623	67404787	86748211	23952228
42803175	56884020	97832440	53868459	90716973	2643102	75704738
66827875	8489288	44377130	42991719	52888285	31435094	94805133
19298661	17603864	96617745	58883745	43813722	31084539	31007205
2957500	2076640	34606899	68137770	64816631	51221100	20412028
40674810	43562311	72016784	68884628	2984400	70029083	10420139
4230121	83394597	36305319	65990893	67592650	30510233	31604127
18144846	14515358	50704675	74592879	61125484	97671209	81576457

27975759	85841616	21303421	18319392	92944880	7946592	78261862
62254029	35907471	13939919	32803419	67563063	3485345	69579205
81013887	37640890	56848714	44140349	92836883	46547131	19643435
27936447	53671846	62594875	95010624	84511331	44931796	77407433
50726092	5999879	74837755	45226369	42734098	5743202	97135164
34129366	56117299	3538806	97130567	39025879	95739347	93282364
96415915	67342072	94876147	30486379	67773115	7015562	59304722
13578436	213556	81913645	65008263	73437156	29415761	61726211
85681771	88001892	94485381	72177767	34199980	29340016	78377704
99855654	61908604	44883641	64904901	97332431	80649566	68062503
94915056	94566113	93909876	32626470	55750903	30978254	34138623
48096567	81194181	9335705	98949709	77545372	38864242	41468866
7107297	89825197	14224096	44304189	4306643	88601400	80746059
19388951	46809561	20258432	14732532	50813984	62628134	91117218
25439228	2275692	24495366	21352635	60613347	51707068	87843345
96023613	37091524	37313040	94909085	90523476	62179239	79424694
25516227	17267580	31633471	38333742	86351097	17688259	54869605
68887266	77195704	19091	190500	60141147	55494981	79226335
82493988	67834866	22823056	86928366	93861977	27325641	50557796
42969068	26108066	23679455	19657934	33164118	14075308	19401178
57950400	32094877	81855521	91877494	35110775	15924043	85114348
97363847	24909521	57731293	42267987	33545756	12765351	78018910
72716483	22568992	53227332	39929400	11911449	81316354	90821106
25977036	95685177	45785659	73012854	38476372	72237109	92975873
64251650	69050882	25954325	54702786	73613307	18778861	3551629
55209586	15505967	887182	39598382	95068467	10800049	88526393
98032359	40776079	73854522	93669452	26781921	55635714	11746599
7215896	57886978	10612842	56575179	23797628	3019404	59815877
31917292	20588000	25002344	56403299	19803782	26060446	26227358
24117560	93048213	86029601	75141062	47381412	1245713	13140461
27283241	65735206	70031263	47469105	52275770	42124830	32834362
61140098	52228549	70504901	84134511	47562336	99905775	69471136
65397631	21149430	4733326	12304530	44939637	29552279	9889280
18030938	31320352	88148328	85765343	23101400	10177371	19868799
59565370	98280553	43562029	64083658	3315717	10016796	66031162
92969330	14403964	4014920	69232221	20944977	1026299	14677153
38862930	75117163	46151646	34067559	62214606	615088	96097761
55294127	66015619	92087294	31551814	22888623	52388351	60326075
23334155	69366847	12612076	61106893	24094893	29673796	72920315
73903270	89044317	58485281	46796723	61757438	24624844	39904314
77708630	68179107	19735273	61354482	97312835	66252580	47305828

27268318	48579713	22785461	29322356	59339408	49828901	89263963
63475133	27844692	22588738	71844066	41879769	28959516	42084399
64655499	81714131	6657005	38832752	31466468	37883559	38425003
12901725	22131021	80409544	82123993	5741017	40378772	37022772
11784762	7062394	34589076	10989613	52566188	97119144	9726480
86243221	44827292	90327049	39289628	12749734	41340049	18486169
87037309	74484027	85946395	16120010	95063398	2418422	41725702
78992598	83754180	85640366	13385116	64728995	1794548	17738227
3661208	50840388	109966	6517752	13729608	92182431	23558449
28074007	31148100	74252820	24161944	36881156	24893221	73600531
62022030	86060478	4238055	38155687	43425327	94478767	64561783
13359723	21470538	21976583	30780414	65402434	22048038	27661852
22728208	48314433	76212977	3134827	6822148	13465160	34797756
68551795	35082466	24851409	85529626	9606806	75058825	15619933
41931415	94677294	12310139	22836454	61183086	37591065	86200675
55897481	47399029	94187883	68795062	33481484	83735858	42288058
61305278	55300197	88934709	62038661	17047109	3435005	42259035
50146643	79703744	91026134	92055853	86588567	96305161	38656576
41400076	45400040	96548424	36943068	21749737	91678107	63169195
11237231	39148306	50152460	73892230	80716049	5691435	52842594
88248958	98081427	78962812	51131093	10518422	84738940	5498555
64625736	49738073	82902322	12221184	70521978	27667859	14039850
95650803	18312270	26808967	11178068	27912067	38910755	78336250
7427881	59836534	76161963	44421623	86693862	48091362	19827642
73991533	55701427	10891355	49131236	99419177	37041604	51359731
25758096	37659170	34224443	46436035	43693533	5412341	55202396
35607671	35786824	12042855	19888135	66648125	34569602	92653284
22348429	22440754	69558905	46812217	94976106	96886887	29008488
12969851	59579909	52576554	92087913	48965308	41989428	20079397
78455469	5812375	34531439	92806785	5146968	21970891	4872766
42668293	20526609	42802419	35616489	15416296	28430442	12637940
76300514	49776345	19131601	91831914	95047252	85075643	82269992
74690046	70337516	72938036	81205909	70828291	54824690	13313565
65146918	81717570	96077319	71535142	75488917	35223873	26270986
19480840	17272868	46175931	46156144	26386012	58132017	41007872
17906394	17352527	5414005	98401164	81105786	72774956	56898478
43485822	67738327	90940695	30135685	47502371	48579285	10745113
47996426	65321351	22750488	58736397	56285565	39519268	16944417
67329178	582052	86009559	48197358	3925022	90389595	53981614
52728843	98142535	53201478	92908271	41543748	60599236	57394478
96563533	59331048	83083727	15170267	86549523	71674675	10121123
48132244	51208676	25895608	86682207	46198710	15878457	10664307

68697580	3920225	89659777	58306322	97520268	64653666	30148809
14049018	85126354	32459305	50743392	90024314	1912089	66676486
27105082	19883744	77339759	25992711	15087364	36052363	63459733
72166955	30351007	65068678	28180586	76601667	65991568	86581516
32166595	3608271	38030013	3672186	91234233	58507980	14268504
85615141	92637958	87602376	28053408	85430667	3352827	99725025
10977391	62524235	12745739	63171818	85026561	43689416	23812048
86811328	54707388	68857453	43813063	12545779	75110800	1367657
88183712	7433975	18794527	37505645	10279513	74719944	11795481
40170141	73467636	81562216	28004039	76098432	2019958	50661206
59697550	58626234	99700074	1741169	46542469	22711391	32614933
55027816	90439400	64047279	17130811	76409357	24702250	69979498
67629028	55987470	52643942	2766416	95729094	73283969	14979330
83884966	6770521	11000149	80525842	91786572	27461335	47300666
11428531	52024653	29446998	950600	32622702	83968557	76622259
77168056	27541481	82696299	95569677	17805060	36636015	9404981
34532272	89731844	72404333	99643151	98099160	40950525	15253151
2294454	39569938	30679409	42536980	84571073	48696019	73440174
72205751	16129826	74540959	76925001	35106374	26334360	81470817
6935799	97140956	62850179	29377180	50120681	20438375	34689369
59041377	84043581	39687843	61617560	27813232	77090990	7031059
92607784	44064207	98320109	82595923	13706735	5230003	26006479
82053808	32874254	47002875	89791931	17920580	13774991	26704018
11579097	62723903	35334665	6086907	65245934	48629164	71746618
72207116	54529534	31586145	49152350	25884223	34838779	14637097
89035883	89658365	18584427	32625109	17463721	46477429	72076787
12037817	12378589	69399999	5708583	67480793	64252311	13657851
86805234	17490815	3370102	48828441	40593715	43927181	1648432
68002995	31469308	34834539	29576611	23300053	83870270	61341316
77336686	73761865	37211592	59581308	9771947	23127867	71831452
89680266	97824875	48765061	2439930	54008260	32258068	26758253
60904360	51102772	50228464	99762455	41883627	92499079	73020471
15224999	15638490	81693172	88134303	93872076	42991761	84633950
4746187	25546669	68345032	32350055	64367904	44254694	88325421
4907447	78778100	89368356	47432904	79943729	27776734	79187654
68075292	52037598	66217048	73089130	36762787	50420633	2867481
2129666	49920991	39972905	27819517	98870572	27212182	79993903
66397405	94940821	43785561	44175390	4856357	18918809	44102976
81721892	90769046	32676165	6820912	86405061	34251472	1500151
57362733	8830535	79905185	37288319	70555099	19892285	49176762
41545527	32836000	44596396	37291542	27005693	42479043	65320154

5728700	96534370	44233982	42533510	40051178	82284846	89904455
96399104	50515137	93986818	49648567	26645611	52931901	50465776
64855199	62460919	84703125	71705180	27100943	37664555	51495673
46532316	5526160	35893685	28770116	18057552	79510485	23039876
12418162	90312840	42220136	29595247	10690650	97959288	53911649
97763570	39120292	67378664	50527817	65340699	30170824	96496475
51926767	94324174	10572246	12146791	17369259	73459660	68515204
79458927	49023052	12396241	29109048	23316719	15841509	44997768
54260738	36858409	56071120	46624422	46483479	18780084	33853346
75846304	22781807	267582	11331552	8199511	79752151	24922793
47922508	56568063	67809391	87995357	77405350	90635856	2389077
99824252	49768837	54596473	35642760	83374346	3482100	96163451
78651686	22794338	84349886	66619620	37983165	21934273	48489673
39948140	2227739	65958743	38535703	12566072	41463521	85382383
37232508	93567484	64249508	69022552	69988983	66022313	11965132
46422247	66986988	37887684	20250349	55779827	39351855	4960256
87987624	6398518	14923140	34847743	68254032	3704332	56030296
10789017	46613439	5948976	49833367	29251666	8865151	14824597
25956998	68854000	80892189	96499446	1288183	61425091	70394894
76463361	67529101	14795678	17724055	55972622	94285025	34555394
95611924	35719260	87962252	85631030	2750901	77984608	47268573
48123446	72141838	97199597	21805856	59785679	59041190	2064741
29644419	51803075	16462747	73145780	19808267	68687488	94254648
67956769	73232736	50105294	40063044	13441872	50756430	54151639
3975157	24375075	92446702	75963822	56287378	90521957	19447256
12023486	11877035	52717699	64406922	68153757	15934659	93218948
57249390	41529914	74079114	30638351	95467070	84090423	15253513
12478112	50574341	31960504	88961291	4527927	5453173	47810755
45975921	56688544	26771399	28069903	49913518	98180508	47702963
54419508	23380494	65107298	40883251	1772432	66667200	34309996
37264351	67384249	73425378	84202938	93478853	89998472	56898286
96505436	35579065	36359090	6494361	26790723	34815079	19005530
67789064	53034132	98725554	79927619	80767024	72298171	65920484
55596793	63729232	62530333	24167856	68977768	73029850	45801213
81053858	88349419	70763870	1153551	16941214	80237900	85332531
61197089	98772395	55465158	45562377	99954035	10956597	63295586
77164222	53902447	51853738	53163512	59005792	59899881	90916896
13483271	7419235	6787127	75501216	38992820	647557	97456381
77907188	16799545	18371996	96387415	70896293	88785253	18343704
48664924	3992751	64186579	43556143	32718469	93698135	36316866
6221629	4798879	82069877	82268623	36177811	59291	57341178
73541367	8253571	54210796	57546694	83684128	19953924	38353338

99426062	56370491	12709037	43322896	47763171	3339686	76794762
50432276	61968876	67145533	68835791	52887351	5160129	73601187
80476395	83705022	44921784	64188766	64989391	79407488	78353895
50217682	64882414	22757522	760657	57921010	10611925	30778452
10208128	79807722	63309269	59436463	95792049	99481801	13788034
55212755	58588022	28914379	49882556	32474596	78251334	34519429
23381073	81395525	36330932	72841233	42331938	87762716	58354744
72190803	36158894	38936693	2708731	99632299	86796174	31053483
96874645	87273960	50191263	47662230	63484322	17911007	12906120
11551913	33303495	42666777	21471311	43722352	10832118	51156318
29373461	79716786	55265985	79640972	69976395	32431719	80009945
57195830	11193518	17388853	87800383	95391033	92527473	66484295
45665655	52545012	38274145	20401632	16653588	73109698	78503740
21578179	97305968	7290238	1766773	41687897	89217231	91003944
22548388	10473639	32878618	35551687	71477026	51202432	48992693
97871444	77373679	40694329	55793159	11749886	99684643	82924479
98663336	74462816	26116787	580157	80431073	16681738	59709399
2322879	24567463	36294595	32742489	98331831	9701990	68387647
40003137	72989322	47570923	98012259	12200739	32241218	89610790
22398253	81895017	49710740	30844489	3756600	50020595	23425911
585361	92610474	58648773	82694395	68470733	36375681	32124433
25481015	58252558	76582412	61187476	96778258	60102757	49262597
40306970	30379650	65537284	57505236	57152310	7147980	26345736
29524103	22558157	16856314	4765143	15592047	6540880	5096055
15924033	84833032	13450142	14969755	82959832	19459886	98890824
31625645	86610871	52664939	56575850	18435127	36334269	75095917
56546040	51915478	45181385	89190761	17447695	38740408	10190810
95230788	13755554	16779965	4425117	66027811	32415239	19053466
74130765	80213806	78617092	61333432	42012424	91893101	71508319
45216842	43159284	8489700	55967372	80811568	6453047	72519109
86368831	88783461	33089171	53175862	68388162	54881128	67334824
25195644	59932604	94415431	27645790	64291012	28639180	92707649
34859576	62820199	58563785	38515510	99511971	57213676	65998923
25054378	43810791	82848388	71881874	31739531	69186980	52929476
77053411	23613447	41791889	61894209	84101249	51371347	38697192
98697214	87982904	12920090	1897940	66104525	43023319	75883436
51338225	46444635	89051412	6400515	71102053	98170599	37381857
17579219	74401765	53481625	80072583	38328166	63706721	68382323
99392957	64596172	23360213	31440129	23778435	53433975	81776535
17354296	55178901	45743093	83520409	85944081	85241090	38947320
33228365	76619238	23421953	23458001	98023779	67841941	64156608

41929897	51973443	65219990	63038479	92076292	94635539	45632466
58407633	9913678	20236181	59863732	35980328	50261936	33319667
94958706	10049286	40504514	84692318	36121849	4949943	66301329
76518563	9839945	56975904	13890725	61267424	64146380	23403432
39302544	94155259	90552068	61213131	13189481	64119655	39921502
58163100	48299970	71998045	78596470	80089650	75564349	9767172
23016946	38447969	78520100	79158860	63358806	53771395	44500640
27871413	92767339	86673319	64600022	6193486	18396100	3782177
98455277	84114433	19605893	45559705	24786255	17260014	87224831
10288499	59075533	8563663	52716152	20887253	99445508	66547729
28836326	1606191	87653338	49074257	63379605	24838580	12853817
16577007	66243128	49840887	68216245	28900715	68144816	21669418
65350266	30870904	60899000	34534833	54067892	63952608	38233527
85709650	66977914	51056509	40153465	6997353	83926341	96975893
95645452	1997516	90419260	83200771	86586141	28057768	62619106
38742785	50682758	26292794	96644348	51089314	23488288	10523097
82036696	88356297	30697324	9644655	66071984	67115252	22287535
54222261	75666333	60173516	26564403	78719308	46064873	8918625
42741671	71505218	38983662	24675177	67075853	51906987	74750964
79133030	7806816	38445863	19274837	80435912	18593085	7753516
15423697	97833323	78708710	16361056	30982011	99514969	7334648
43133989	8117130	43939714	85786999	95449594	98492379	96997800
62166917	1221906	14498807	35844653	30274730	66754131	63903732
5448072	12366933	9932892	43735812	63020216	20068544	41578081
86919227	3077011	9834791	46203086	4764848	93764169	94397085
48413372	40958434	42471304	86843839	29771505	3063837	23021929
44410275	35891305	96034507	75122310	56727542	60245034	25218135
87163351	25584857	68894822	21323028	33025826	16006684	11233724
74708322	87502579	39170992	17258529	45449324	6593866	7458695
43120381	26415561	84087902	78551341	18495775	88834524	62241163
76618791	10847104	70084106	57635105	20694984	81849231	49147117
12889004	66921431	72709016	80945385	79144881	72759595	27421320
57811787	24995992	11928759	31404378	36019608	81531492	83080570
60576363	85027790	78263219	76063106	81982962	53124993	70454098
9645500	89843232	54254229	71429283	50290141	90008033	57429577
10214643	41343918	76195334	43678962	56907411	84771326	81363223
67146331	91284851	6081049	76582758	70921029	14451925	11059912
16927433	82033227	76127341	9971584	2102360	87064407	60698742
38090583	90990298	41317017	65409869	96989159	87516748	35115218
6695255	41126705	5432069	73312760	19611421	66853762	48116293
5133843	5221162	45729979	44463652	61079608	15062988	21403797
23490036	59697229	49595973	23484393	35168031	16017305	24374131

PX 71900-10-(174)

7773907	52185348	96526545	92539243	31624966	67509478	45973918
340840	58934362	63721734	99992772	26950666	57725143	3476041
76277749	80447875	86696610	48725742	12085447	10112003	60007012
40151916	97639149	1560414	7830152	23837507	36737135	91320172
60195115	97309092	95173526	13493502	26798734	68558436	87687054
43198245	30785259	67482571	20813604	63382864	24906191	35814903
94203504	83189194	29341435	82513394	76886837	35231744	63538932
77 85640	87114157	53838295	47460121	67976443	89823991	3862801
84840397	41079517	20132846	23144481	2339844	56995796	21050759
24058938	91941989	58332433	41896923	63884370	97307174	41217013
22021214	78364725	3170369	67153309	19152079	29705302	19712120
81110930	83266458	55421925	37306119	34427634	64034933	72470591
84542110	43443142	75038874	84929301	45906450	92331730	62836736
53856536	23738715	76127032	1278903	7305971	41708906	9146516
11241556	26599982	37664327	45081656	35756919	73630397	96895649
7062587	40018485	4129621	74735629	50387295	4538486	34059855
13964207	89444414	39586618	32563142	19059964	26395168	47269987
87901605	97155066	57134638	34591793	45834185	95952231	42263697
47078687	75025258	10848789	96598946	31455823	6857555	43491445
91704891	11341056	7127297	17209075	64563025	30404364	89694348
91321647	15019926	17958052	43808038	5401944	31153043	79089859
1294720	49611163	48151037	43015944	2630213	38274715	36436675
3156091	68142733	70216304	18487156	58426285	2873034	24127015
5482099	4344417	77504685	48843738	68270447	39143745	56061713
50425406	2922139	63354619	94017692	12796669	19521963	35228786
54438403	18057189	98715769	73094838	63120542	11543956	77895378
87217892	83518406	29596378	42507551	61990986	18047770	62308328
87033160	96083012	97470922	84316389	63817692	919640	40275970
97822463	41634400	60653129	97098715	82500522	79928208	97336568
62812492	42921349	25870935	63678540	52126327	71132529	23608338
32284995	33600562	78153788	83528929	57190320	99823674	40358223
6552534	52680712	97643249	21213566	40686427	96206055	69213134
18673146	67634743	6696405	73478107	28466853	76702267	77463702
23053412	2661832	44412802	94338805	19513296	28295502	88153380
93668733	93979086	27009603	18256083	56788443	93447877	54949932
46917284	97894612	58039206	46734602	59021406	96494347	85693841
24897775	67494357	83937825	17386627	25179320	68975014	7027731
6323591	54586326	18626320	3184950	74687776	46380522	30277124
34040913	97520257	83521827	81805499	26853163	46525109	71216784
16027454	41446835	18630783	10421236	35090546	47497359	41801162
54323583	9206115	77247285	36415801	87024579	14013039	12501660

WS-174

14067492	67959624	19330881	93645418	59278121	70554668	7767542
7345096	68618057	20093951	85230382	6149022	76872711	40497179
12564362	61010420	81956011	25990290	65888939	18030255	12106432
63583856	6853103	52467048	81894482	46625974	90143807	99567563
10495866	4874608	94486800	22107905	90064182	26381098	54110795
62925421	10175030	88230324	37032323	64645058	56426985	25858733
16601757	91412932	60711130	67872392	61031002	10796392	95488428
13837656	44743671	19631361	90928037	34642553	14789694	90269476
30278632	42306118	32492042	32165263	6236288	14527615	50666737
81076002	40208132	68475491	36008364	33654628	41262332	62747562
95381227	85103927	72639334	98066877	72300426	11778474	98606008
72344074	15070531	99383138	56806914	76196752	83569648	98736544
11407224	95544836	61759788	90734232	85514140	69885030	94442470
88955851	53270077	34457411	23542601	62042629	80878448	56424798
64059779	26242710	53341533	31366981	57721956	48038187	89584228
80471620	83594101	11538302	53056732	72507385	45876374	74839042
6776045	32181680	25321092	30724067	90880846	49258935	71506580
77298975	86630874	78511508	86435724	91546020	39457823	33225383
2436162	82225915	42690672	44332406	94323013	93234670	96882075
27856695	91817529	93026589	14581621	35744292	85609610	71259506
79301597	52254906	64393851	8885932	62557593	95999341	25368651
26460134	18134230	8450087	15329974	15968884	4388578	4022347
9730562	66858906	58607795	3657290	74838295	46557085	31230466
74061808	19387513	76184528	8123762	96291118	46256939	43343819
99490862	253498	17778516	43811495	60091040	19164340	74320765
56519003	29847712	17652368	15287071	16996071	44046641	6812654
5191962	29592042	54746426	93319149	57496994	27718189	87728731
92967659	1706116	80136606	54337028	29416424	80377498	31142620
54309082	3922180	10439434	10159479	53408152	26421466	89873190
22169301	83547447	41261382	36022491	28417324	25546842	73211808
16506105	84918495	90621806	9792518	64953215	1240858	10779007
33451035	32461484	99415258	20871554	82999494	61475735	18699386
45386483	60781533	85436998	47236602	88250423	5076572	4705925
46775439	39079690	12422503	96813913	13348473	78558856	95687429
6009610	80151427	68623093	27547234	22154369	66138769	66839592
52143936	29631918	54082833	81614471	21628626	40068524	33385610
70996194	80593361	18708899	44567455	28439703	74713664	62573747
37515944	31571964	52870545	69250459	93868295	70843705	85792825
32109405	5371698	31198731	22191601	39794698	25079037	66420614
76759746	83830323	80185271	81186705	67458914	26391698	83870875
13020564	9825486	52873498	18105506	32583383	51601866	81127065
299788	29958275	22721518	49068740	89715711	21209538	61589852

10-118

WS-174

48725164	88030870	2702965	71642908	20117030	63392191	44467495
34971828	1757624	52746847	79082014	49037348	11913187	63341875
37173087	50113783	94821680	47595336	74100740	40867254	88621270
40533127	84316975	80302842	33210232	40221570	43855477	83254266
49804467	51165483	18764616	1366101	44410858	52292059	35827379
42520839	33151225	54027073	67195314	92498507	56929165	25658693
86255866	35552688	99636812	79536602	86665652	32785040	87402301
36757251	28901523	90875193	24448421	42889404	66729335	37439872
39024407	54329552	42906232	3257901	79293018	79348899	40761137
56164527	6711626	99016833	86806343	48688819	5109991	5313584
95576901	86810402	28657476	70533484	82915218	12387092	40167999
13209633	94157840	28942140	25539043	70901914	89630562	41746410
48611460	19789149	51396537	76244924	96550419	93065718	41603298
16265703	32898939	70344121	14755921	1805079	45557222	89153085
31301311	89364410	70643124	80262051	93651474	11108048	28895465
54671359	31607231	36136938	26775291	3340574	23463321	32994660
78779642	63191120	91152840	50489618	64481461	17575489	10410919
13291724	62988413	25814168	77341657	79386591	27357627	10852794
95935491	88382689	45638259	87156579	61189296	13760141	11602136
52643113	59372596	43938148	41732828	11023310	60994447	35262281
56127051	9443243	37163805	57431224	56547478	25590200	84984523
69050484	61735903	89898309	57719685	43812368	36921671	90905370
68070985	65092291	21364699	29270234	57806171	1226350	5298150
64046132	20218214	91293212	572026	66762110	19930745	57378363
45875108	39224367	42390569	70467033	92586850	73874971	94220262
40145800	59803679	91420557	65618943	73006618	57427574	88084668
39734695	37464547	62084337	94028998	28282193	81519082	22532814
88014223	71266228	26686002	41630125	74608035	84835653	41840677
92204079	95681589	79067011	29631323	86044115	80790708	67479631
72324268	80717837	30763771	55654050	46311843	23893163	13038969
14028702	50474577	44006432	83626518	41031206	3241775	62656296
78054498	123972	77645670	41840617	90516179	65965544	25571985
78355221	4242583	31318089	24486394	37390889	3228864	30095138
50350894	54259962	98490329	39327909	2408302	15267875	13851747
38493627	20777892	96570686	40872565	59030528	84675842	89640210
44725155	35616562	17469907	86282114	65209899	8847468	54243466
37083744	86884225	92154671	47930094	1539336	85959782	49075623
14910431	24690309	2494871	28314770	93323183	36762360	38408417
39598734	78276323	66412698	94656933	77363407	20161432	13860263
9627339	71195301	44658077	27649647	38577074	86578672	86377898
75417714	45431612	45196917	19506984	81658198	43781921	75993897

10-119

WS-174

10-120

23677817	99015461	48209713	83056155	44830663	50941127	15492401
21906344	68107068	6424949	87382255	9701021	41128074	43944304
80693564	19012766	2914259	39749552	52763351	6495924	70760498
26743642	52525943	38702541	74966601	32897823	38949195	85975217
21624182	49269180	91279922	1637853	38833681	57600775	48019070
88662305	49274000	92656768	79887881	36613169	2168349	64395950
27928702	27605317	74327660	88371892	66757288	18497636	6969805
25290908	98582122	71480020	1210877	55828940	78804512	27075241
30453700	1202636	9272740	72470560	83670028	6769	56205439
67143712	17608048	80102858	44474055	85297752	22330380	51563437
73797546	36446166	1718166	16469856	4698255	99441106	76929914
39099464	18323481	73757487	48268938	4322917	9545030	82028213
13301422	83399524	40692251	26796253	56165399	14494098	23498064
17102693	90709869	71231286	68900504	46949360	60721459	6068733
64225570	66695651	29377325	54199770	15802530	22195839	24798921
8620817	5519727	71132829	32047834	18108075	53828121	29684126
95084848	68971494	30749578	3241110	43948746	71436645	92039388
26054063	41757208	25784402	45281203	36637643	19573816	82704726
90663397	99729738	9344149	68057942	65549281	95794498	34158494
70229757	89852370	83229476	88798910	60484668	55632581	57827854
74334098	28945166	21390283	45972044	81839848	53623260	37967599
19098741	5614133	94800604	52116131	52737638	88453065	68586687
77128051	57985285	73943541	47784354	62381607	1009007	65095499
48050680	38379801	39889682	89138052	11051336	44105822	61059351
49 86016	76520115	53500273	67800760	17172204	32903371	60805885
99410403	18510101	89195420	14295344	35362374	68881476	48531154
46152028	44813851	90517689	33599084	36575203	7863786	30493472
46312967	8891410	32445067	33783486	58200523	89146789	38715650
92636686	51818912	12974616	59409514	8760737	30276061	63232769
54848558	13671250	61088086	61989821	28041412	53720924	67325703
37042282	76762283	20982602	50053437	7801606	26097672	20445200
34333950	40041358	74467457	22458067	19731324	84480307	56939655
52324291	3042765	61387630	90296816	38887368	91509891	28595316
66700317	26421895	25328599	88304487	49591322	26861797	20978356
508692	83268319	42899232	74771777	59128247	15100680	8000287
85843018	60744000	27324320	13395881	99130798	71604853	23308750
60095812	94295962	22228383	791382	57025507	51563972	88847981
38155651	42412588	74879259	64404962	13015715	7632822	8305201
35466319	85853066	74975406	84824567	14000859	72511295	21653414
66307331	13463806	74466260	89094892	4848689	36431093	78819212
37957376	30212328	56113103	19716112	59194766	7342000	81139158
72060427	88674865	12805566	1373835	93063666	46881750	37788504

WS-174

1356265	36143362	73275310	39822792	99859792	44099661	21957957
43671133	46351924	31074443	15425479	18327967	65738360	76961179
66067778	83011242	98897243	77985006	58464971	12313736	67110307
17394233	30859196	22365512	40358380	232989	10476574	18877279
83754925	98208743	36693854	22703496	78950978	39804819	64441835
39155881	99746913	89852128	95563310	48631364	30281963	1794930
28484519	36805160	34483669	91134939	83775786	22364248	64802209
48159565	14487235	64864529	56493797	55142424	93326563	93582069
71896502	40596186	61098195	1075135	46337342	23511844	2103706
67601856	21054787	3155514	51910768	46899021	87596809	5662217
29433270	17412434	80084283	58799489	27617508	73977933	75764113
26863052	56283582	52106134	39940946	68284074	19840614	71981061
3461638	73750716	52810783	86075074	12194497	24079805	7876580
65465680	56432181	3320326	71809514	9398946	21388574	62026180
68589115	11213980	56147852	57738872	59338679	29320925	78187204
89585696	55407314	28927702	22027403	88773070	78268944	42220895
50947172	51330523	98210089	74559061	49293295	98586125	49873304
40635986	25120788	32985761	96871507	52681347	25840946	25344728
24555208	94194355	73082108	53763567	92721506	22026725	99180544
94739403	14708592	66605710	49325037	52017000	82590904	20470425
72866290	76138159	45864364	5413330	83024602	96716671	72389028
71741366	44594343	96813458	9046984	29814029	6197736	3737903
60913967	52928119	78289587	36897777	89816263	31331353	92520365
742224	76241138	67073468	54794044	90714983	80878838	91747310
79927012	63691110	63837232	13741253	17122361	84919593	30428699
30741858	54500195	11537908	41972861	52662286	60054706	39685756
2906798	34440094	73257655	80029347	14549904	83016708	8863120
91906935	58028441	80033662	7023262	7483004	55884126	20665251
50717122	22094396	42617192	59365951	81573032	63840555	78053099
94985515	93142166	44728358	3523897	80368917	12961098	81778995
86558165	46352975	60640811	19008454	15828294	75891141	76161239
24054305	95825763	57969049	19851186	74761251	18732247	30296679
13127171	56066739	57597373	32973228	12732940	89035838	88392440
32090045	68450518	4565202	70190821	24851038	94882510	14013403
22741582	76631340	95435778	9101065	74807017	45383912	45494287
63517886	22597756	57099869	81481976	52421474	52606764	30629597
79921767	50358311	6209011	52486849	12665472	86829698	34614966
26640370	62106442	44790576	58174133	17841072	58784797	16960427
5506744	26560397	81042338	32708218	73756002	6493431	628284
92145920	70185443	89840269	35571206	83707874	7264237	75621998
20916174	18111831	25272887	16814951	49115078	51103028	6632611

16710929	75594151	42840365	29358419	59213626	1034088	99576880
4163033	75146366	62373978	54828072	55627959	62021076	59222880
78466468	46796984	69099800	16330959	1490315	49095237	29813787
38459900	11505684	74982529	16770885	17425152	35209478	67918133
60079965	79174572	62911417	7272962	52635226	26006373	80806619
3787278	52842549	86983033	53214881	92315357	4890344	295600
98934843	61993383	85479345	30568215	1473556	67105971	29633108
2041389	63401052	91307258	99556115	91097871	45236682	64420393
44462088	17081701	37041094	95028722	69428163	5484013	58188403
16626595	38371158	65183250	30055010	59682537	38938892	64570239
99128390	3863829	96333615	33808300	16587357	88970294	39996060
8119333	29628281	467622	1643101	52250874	85184749	64594522
70760528	27587592	37697731	45267340	83773578	94467284	9961452
20142150	98970707	65788256	4223194	22744356	20450821	58244251
10420847	24147332	28474607	26399659	39391519	47032368	29465665
46688066	92064021	47932464	68211357	91672992	53493512	11820373
69334640	46099982	57068780	60945496	366951	48211888	10024775
22058175	44396975	64393487	92858166	8254806	24679239	93730641
90723076	40112861	72699214	73634366	74332457	84047184	6828465
30447990	56410221	56637838	17429650	27527931	88630293	88247578
59259744	90450169	13197977	34690734	10635904	34258750	72024603
20410577	57174765	89688861	71180988	44662894	28942097	24329382
49568608	58968041	71643623	40231163	45286981	3019465	79790437
34753499	31488722	34667852	38016031	6985127	8424191	92139022
10351020	73064228	7316972	82749779	15901538	58082035	97321133
31253592	89127279	39261910	24795443	7298197	91448882	57141683
98561569	30161041	89719957	6438814	74774457	68941344	95776390
61621410	90246285	48966077	20207974	71660528	77298653	77572482
88166148	50200511	22229857	42257440	5276654	4125123	82417176
54469884	98363430	93327766	31472003	59374538	98569820	68116845
13032677	82148758	19620269	47327183	86306968	93305395	73227405
34353173	43945236	68058183	67150248	67258813	34364118	83410737
87697035	6613112	12009602	23950331	48745660	36348029	59534296
29423895	22113631	42287171	36352500	51093195	90731169	90089636
71355587	43657603	68383826	7457105	56795526	24268929	93756175
3730369	93137455	46417862	6971272	21313305	27937207	75051904
89356210	16807810	14168002	76899909	315545	18432807	62270981
62976666	98004809	71055371	509804	67794814	80573416	81231317
94337412	2559113	51380014	14078784	2935674	41463376	81303293
5524607	56863591	60001786	27353619	32318852	26802611	29939423
29254483	88112021	3913466	33735654	60109508	89532195	93092260
87022794	98015764	18165539	59772768	27149359	57509449	88588130

64561764	9347906	39530839	74020333	60586703	22717447	68762503
57067047	12193339	92502432	33128180	76821740	82993636	30898232
24374480	74460879	66701750	66734556	50097511	39716585	64871200
80654507	73008619	53556544	40184343	70340207	14422590	41883243
81696524	48213880	66063031	83688007	40185226	95180457	74256270
75257561	10956589	63070532	54256111	32081733	32218928	78080407
57898363	44602990	88573942	7907722	58526283	11895132	24945335
63258555	27269114	70972510	85411901	12305114	15692619	17225328
50974366	11036362	25515615	51008	92762142	91442805	20403348
64 27506	83310526	87168114	25358198	835407	13278521	18825321
98485439	35981847	92993919	35044571	32544194	22892290	21329399
78034868	80058218	92157020	61055832	17293416	87354252	27236674
97905740	98636154	66036431	6474643	8961675	45849836	77115064
13183865	40350012	33262762	17982849	20674508	42696389	5607200
33981573	59205284	34795284	30255602	65522759	51546141	95638250
67828798	97075460	99387542	46480992	38092143	657951	76201271
76478866	40996080	88831111	13801333	96659416	72323897	9201615
84872568	6578872	79825130	16154195	38322960	51471056	25104023
98229488	83414111	48395509	75313605	69463231	39698883	71257602
52457808	1000467	93286627	82123429	69959962	54919942	8555340
23480968	73034843	89826828	60106824	29456307	51417344	97150402
60146330	67083617	71866747	8845180	21794263	2090867	32182760
55703279	33413630	53938779	55668983	8134510	84185825	14715037
30860697	42194660	65772614	84656407	99627835	95434198	98870779
4597546	84902621	46713110	52743743	96681511	22804159	92195406
21440409	99514173	68325789	9800098	9755699	9504581	17872752
10471509	78547170	35376373	8192649	20930467	26742364	16573689
7138779	35941841	41300781	70639714	11398854	22467916	28364531
16787424	80041399	60361178	71315520	71174225	60813828	54424677
34575287	98810762	61050514	1907006	94360062	80636959	37233211
54732391	82320715	25216939	53517893	26609597	39714727	12602586
62572950	15095015	70182330	2266429	4976316	2911080	20682110
23146192	50473105	2596637	33250386	25017757	87343303	1071931
90421250	4965035	88211071	19415726	64556522	32457249	67362491
677616	3541682	43819477	58764481	80239550	50361024	70798164
12608709	79660823	91546952	32077818	56301230	77548448	65119465
22359058	53016651	18704374	51489447	5602365	32182752	55478226
10533651	7491564	28255630	40229884	9306595	85343621	74921318
39634658	41830534	75300165	94026981	87857519	92108497	91156368
15520366	3301688	3277636	26779506	19011451	72098413	83542061
23962020	49832878	87862476	97339379	7672844	27688257	50998423

WS-174

47608279	64342796	9013758	34635779	75273578	17179328	99095548
19488135	88275558	41077063	85315494	98595104	34031972	617520
57772249	64914516	64853361	10754938	27438888	15157778	3648748
42834919	10371158	56953910	50687839	3621531	8402565	20634779
32297059	49897761	57563216	11604408	33679679	22144505	57083587
74837717	37174603	92761371	69753300	12538306	21279235	9015841
93234015	78455841	16277350	92112861	79705180	20922334	35129891
16817834	61256921	37115070	28612332	8517635	36518766	30810348
70491429	22073781	80766388	54406439	58381088	68465283	17275893
93422332	7415463	34892255	42610434	3469959	39398756	36785868
28638706	79373245	54565753	55189395	72521217	32340232	56466120
18556445	50751526	50411734	20959346	2596856	39411218	64326914
99097736	81040194	67765477	60581442	93690734	41814972	40168545
28569516	11992277	73440332	20888405	20175089	86075172	14951400
8776117	60287720	18086378	20253924	22132763	7372489	33922095
95871381	49520823	56874220	90577684	4354655	97081106	38283388
11987057	60810824	51325818	6557662	62722107	50378607	28566130
50956223	37514670	19233735	45122289	67581535	37892763	28913293
19331580	45284907	78892142	76333003	24221566	55522787	56148747
82916702	14101189	76387882	91663614	58110314	91944339	36523014
16095887	87532141	65492589	8120499	4497530	89859373	36127417
27369267	69869794	68481769	78401162	88679361	36536315	30402325
32333905	12694871	12138031	12991904	8878642	9057882	67957341
89324042	8766327	53314281	70029506	29343696	47684464	70882947
65046924	87293219	55107881	61098925	21611242	87898758	44270236
42791182	25317069	51768150	61350588	84307623	85651067	29553946
56784748	24256013	25360618	68914011	24270431	1792192	85677777
96548051	26449961	31335966	98658274	66278110	94418322	97324884
2557699	11601855	96077294	70831851	20755651	43681838	79065828
84815046	80501948	79854923	48974203	44742146	60623620	72268097
25195481	46593699	83524433	64928130	53250189	11845983	93068573
81303985	11296130	85461876	76007258	8461476	67285929	59847944
12322798	80010153	71381545	49518508	48396408	40867271	89099508
486677	820862	6756793	27461983	81496475	57649145	13010226
34178312	90871970	67997827	20302646	50585943	89908501	76002050
67961170	62822419	53748825	80657843	87863767	26791586	59092696
88734062	54651197	98724402	47519953	6319119	62999296	63535506
71006330	97300353	69019604	41986187	24891527	25945509	14084880
20241916	86981042	34094185	5941665	47794367	75628405	66937956
713691	78867014	40529434	21767452	53159451	9372767	67823257
75415692	86241669	1166965	79530324	44272854	62325352	29075531
19952158	68842699	31411819	57918241	66933103	10821982	34449327

1166713	72441147	61540049	83128136	56509351	26756440	9901671
48599933	54736599	85304838	87453554	2193921	66873659	70980907
53197946	18558975	6996024	46539372	69805385	1380512	47162944
29394950	13149265	6518869	45152666	43762579	68742351	27028862
23172701	20969270	13339937	1728538	39815919	40728660	82056886
27797859	47276253	77942892	81688034	77811591	11852349	29348321
43575637	44246026	71756269	75621283	802042	14189971	23835582
33194709	69344377	51582444	71624987	52840566	31197963	586490
61525743	83329874	94588450	63118277	55224666	77789631	65169861
56053475	87112294	65661991	76058520	87188923	73878880	2011096
69793941	82095044	74089567	13182678	6957739	43260811	46039878
63109058	64314401	15521802	43698744	17788663	60827935	48624740
81509907	32860342	58288485	13526532	16463179	85298655	47733264
64777188	15366083	92250693	64833243	55725338	79512407	77108916
8841047	7827842	93071114	59183977	65108183	65779765	33792773
26885115	5863631	52504	629369	3285396	81827636	60360799
60653623	52899493	78299219	39426415	89190481	9570831	2546657
76619157	21143289	76944078	34903031	77321685	54412103	83501049
78186166	80283974	39293091	96662187	16058983	16333055	48573927
86006775	46078041	91662467	25843306	91735432	65203542	64031627
9065172	4601645	65996544	92346900	86940748	71627431	82240733
56892341	63480479	44018788	28568055	92014278	42850805	54617434
88439454	88339924	43901959	97418215	77976294	81817455	94759766
50682556	18836925	22271972	19062509	60089859	89073703	79855706
71001288	89678280	41955410	98607699	56171188	43130516	4504637
21355584	9355867	61632649	3764764	11251965	50986916	95653167
84815501	11802492	66908101	92597256	16329607	85793711	15833802
96622565	2916045	26138046	61760083	99033070	6709102	28012538
46753906	65203948	75453079	17269347	81342104	9904836	3418891
79166442	8217553	50432546	69564423	67824957	87208006	73844327
75302693	30926075	2337763	77873397	5726911	46206840	62336977
87670556	67021406	66598866	42653194	64325177	50233053	94640888
61890957	22638810	4057252	52141859	25408381	70388935	43044561
2367396	26772767	66554006	71200992	70537594	64318589	12326947
93020115	89958894	17245497	81490789	31910287	40997534	31780981
27284499	63929035	46174680	10963444	25600732	25922073	25880070
52225746	49381180	20799731	75765607	34343076	28335379	1999806
20623020	70648781	66585709	23534732	22307470	78155877	33510665
99339294	31362215	57864220	23628014	48932513	15521225	27466780
21539520	83506168	87974376	24046023	36913295	38015603	12662625
6738878	60354308	12268783	67383311	47037886	967455	6485566

19161715	32346467	97649257	901294	16943392	38494232	37797543
34571731	18327848	62390693	18401165	27950231	38785385	41131679
11141190	21710655	65978618	80912628	7629995	30234554	10278612
49373323	68204779	70842462	50825179	82448131	31028411	62646449
69477554	39975384	97557888	81801319	43480576	54377397	46687951
88828881	37711966	43068255	19602764	91753487	36249648	82553239
92702086	12580911	74781888	9552211	15606166	1078523	7429659
31596035	58938955	58712695	1877809	78306519	14502003	77659616
31511484	96298557	47543906	9067995	84017305	73909255	22786551
27078559	79054245	73155808	33477040	92936514	30759692	19329206
46524897	65252873	54826285	58183553	14405799	55636506	34026870
99463999	49859236	47533058	49135060	76358832	79745425	55622065
30430834	29032869	27759074	29931865	85070896	82946757	54291690
51527361	72250277	60760313	25354205	22723440	51717599	66801871
64731628	75865193	17289713	73902761	85460360	33359670	12349721
62601770	20542924	99117174	90992379	99858990	21538075	42855928
64518171	19560401	77266558	44137546	13983856	99092771	75584259
32997497	24371418	24151276	5208124	81602655	91878183	54493485
91209963	46841370	8180868	92164632	59714046	72167363	9953705
36007425	7239004	36864605	96157122	34824174	6427470	24084427
3683358	2867891	13663645	97155085	57669139	38603610	27773883
43880349	26364078	93936014	96843010	2635444	51213670	32167362
25185250	61638521	34736110	79178298	73310990	90631319	8973489
58533950	73758926	88750388	1042534	68741376	11905347	43874728
2453777	40894093	38340372	4428509	27232654	31374402	34319788
44294054	63455414	9987513	47558297	11257448	71015049	74144791
72135206	44848193	7218115	70938445	66631193	95115344	42815495
812817	48873090	88684396	75560713	41698205	78155986	36577015
58838343	46902980	98152613	68276834	84602893	70585495	69675681
7963457	15831300	26237166	31597369	96466582	16293593	12184409
62330251	32674777	72529447	58664530	43208841	60432812	73232524
44141383	18276817	36196188	82963072	68923843	32425400	36463749
93703860	8417882	48874143	24958456	29720807	53240236	287084
23084162	50297053	16018571	59988805	64829798	18977408	81977050
21028381	65617418	30105845	13360175	34186045	39978597	53727166
8225254	51535824	99541031	69413182	41878618	85805350	74822670
83075513	52532315	83739714	81341039	14024598	69239996	67962253
93289000	14662117	18528560	71606714	75661759	65717057	14543460
35954056	94316615	66360527	35452624	65860635	27098280	96967687
20344779	27897382	28450595	21991016	34151397	4797135	62526795
58855791	873173	1956753	17430460	50314571	16965138	22705790
9267265	52667481	76434183	52010060	21574951	6496988	96275202

14015741	88513390	20798268	34608997	30401777	16917758	9979124
79997305	27883565	42408712	2425520	51284306	6008975	62287826
47148895	36826523	93589237	4607867	65499164	58868398	87093482
73320223	95340229	5504188	38873617	88250100	24126348	26705716
59346646	62265028	76892081	48536411	12588119	58245491	11993739
39853660	28703601	25928759	79750582	66479278	24587876	73675142
13469241	15385309	96238966	55529912	27664151	43945404	11807477
57835379	17589340	90686782	92844424	80773652	82045233	60080774
67819651	61438985	58602914	564413	40179492	68091280	99153303
33619093	62590562	93850626	36303270	8349134	84276628	19016810
69441887	13677241	95407099	7619719	9589131	49626657	81369705
41462337	52074504	89660817	87563243	919205	88953511	54099113
2726891	73632229	14215112	60004863	27020505	56511922	99082923
66979183	86755579	97008529	95555980	35138975	3930326	85369945
39972979	29901260	29407984	41207350	51410010	52609518	43941861
11967933	59691980	36150647	9778956	86084904	20293203	53374188
10482056	1227296	31910695	52475249	88991466	48094228	55497531
16744326	75582914	39259193	82579466	56553308	55380016	66296171
65633218	71932799	91170366	6654079	90737260	36479061	63301072
97276749	23036412	49187840	84752538	35067139	78559646	39897601
56877070	36535171	89656953	40382661	12209047	80071261	95122270
47 72438	65267826	5446554	78146589	40659315	10315511	13661411
52736195	47859088	17289011	97652042	79247951	19499187	9202352
84983198	31776042	22559986	68309110	53056071	53912362	5197748
58144066	69992530	31587563	89043036	22448632	22744621	27905709
16491437	74936574	66157992	70743471	84616877	61325365	83509825
56634108	46716307	81223702	34503904	60578102	36821055	78983130
85839133	85670251	32360690	95112592	44149743	88670187	2572243
18095451	14285608	29908044	86035399	71362742	57429925	20004459
63772668	76497317	83243458	79482516	41532005	93455359	60313608
75270005	50882656	33203732	54740838	70337193	63851512	12808092
72434394	5784715	61716249	73869687	67348523	7917993	79031049
91702595	80968506	40986596	90465957	54687351	42054407	64984792
84247738	11599685	35031604	70414217	83180405	50527782	64356091
61002242	65758591	76705298	28513018	33222808	54509363	1013458
56091764	90503381	33744558	84094716	1804977	42687793	32222956
38049186	166029	84232	87884128	35357318	53686731	78681365
52407338	57590883	43003937	67506624	26375645	34926613	69633716
3885758	59570145	98391474	76945511	75215692	22400602	13758545
38635420	23393884	39136965	89911832	35490862	72601067	92247988
85232501	65759998	14585275	4319628	51237747	38404562	65368934

10-127

WS-174

10-128

48087	10589580	73265277	26013360	43797564	26038880	90628741
70667813	72734619	59099187	2901174	10445399	43747068	35045445
57131260	73780792	32788501	50548228	38739046	79716230	39624786
32550299	60418311	67948589	96136240	63635525	39843883	22095873
31162255	69803203	74214970	33120652	63000159	87813124	51163256
90333150	76703123	55995296	61053515	86330101	72986314	97168635
36199871	52354227	39885600	75357549	97721095	8557156	74568052
33789543	25792829	13911198	15216707	50806268	3051571	80612242
12991752	4602630	93706224	74921113	33867669	76122804	16555996
12039852	69626505	69463855	57253036	9880198	15025944	53036620
43594466	37066032	86467911	57495480	38556153	34907245	61650748
70080368	86696709	51510776	68751507	16166833	70094567	83484885
26119290	70993687	10067304	10509749	82061550	92532318	68564432
81386667	55576194	82181000	54765715	99079717	96050240	24186162
47084920	21035108	10691135	94105409	30366382	94939678	11631080
53884173	26141421	22486689	19609165	3388750	36556618	21907902
67943123	92001366	30174128	55225733	21027447	39342453	8901899
52192919	65427460	54982178	14526513	96077823	85713496	40954528
49271754	29473072	63366714	64846608	89217332	93845241	93385186
36178857	55019869	1094871	64672759	30395573	65194555	79647906
32977651	2941518	71647036	2026771	51396749	82208835	99072928
54238755	38773248	63544592	18300252	91375929	18119027	93490543
47255543	84221175	69644920	50637357	59874367	66725016	50156985
7667023	32260596	97875075	45301988	22538017	70787915	51304056
24738235	8555254	21061645	61864671	88476513	57129076	12341259
92987168	71137208	21018620	59459797	81127430	10567841	64186201
32 22377	58605195	64732716	42859255	23894457	28222111	36809060
9979392	87536589	56404459	28760974	29313425	1417525	14658041
38081665	37744258	11971540	26945129,	83543996	78396771	99370995
77878705	20832004	44134391	59446215	39486907	46100048	58925469
81982389	37005555	34781955	26696709	74400292	12043698	43603119
12053781	58819134	43392945	5091503	22086471	35103897	90869981
12043967	51170534	26639874	86150456	15897405	6831981	34201916
83802898	13129125	7490263	27309848	8927764	31401404	86573796
83425680	96384469	21940933	13375075	50694510	57070069	97207198
47299031	99651005	76155988	10552716	91281851	55903814	37650348
54482445	49071956	45969383	6981516	41058516	74093890	28536016
9105954	78124947	40790363	73033888	93326676	808500	61646467
7095856	36406396	90881827	76856088	36275523	33086490	19318781
21687857	95721804	2421937	84706316	27255561	92357383	39315415
53584598	24083405	86153482	70457600	95656411	41856859	10559597
35100254	62975110	54231955	17542656	90666526	59880546	6503449

WS-174

55541366	41935840	84942137	84621737	63827477	7752421	36924777
44064561	14619855	37588138	49242695	17301698	51905156	23241905
58683572	27027250	97614832	68012853	40355700	59057884	48939008
64018592	15424412	74235010	60008093	70562234	86393911	8608164
33112466	37414109	76878804	77685621	91986514	15015982	41224536
73225381	34455597	27046971	44803803	3526934	31587023	73851943
21117916	34248254	45190215	99404093	30440881	43236144	23205001
94254826	72964204	12048670	49255977	971978	97150296	57164375
65836428	17203965	51135815	97075183	91595073	8986789	81259562
25254043	23572970	29047902	48009049	75190453	94654669	46920101
42923754	93527564	14975692	73961527	16889376	82696123	13801678
59758360	24481668	38658418	93218584	47009468	6828472	30644911
11575266	19173378	92010495	18552544	75227671	67916755	21314545
62820459	64204700	16458938	99176777	22892906	38658498	95469116
7430060	42876829	11196828	61223024	23060481	55199987	93744993
91815394	32965507	63963893	87263790	32527795	98432203	14760107
90794804	44674240	79688391	61774030	88732068	18536552	27999440
80938682	69483476	72353313	6543429	64977456	12091650	50394926
84993412	50677303	60786297	85238471	99488246	60878818	3652698
79875466	90011865	31012348	13420870	36511078	82969646	27108486
84 84420	76966641	68360923	34246467	91328363	22186039	17544164
27770719	89089523	35695280	84453822	37721225	35103786	87747367
80358257	21320709	29480217	67767480	2234676	51235993	716852
33573446	35700176	87968673	80981259	59285369	6017442	32043087
18784840	33429766	51974143	84912148	46287925	3392830	17116040
63529875	91432703	4652832	29576525	20880730	72700098	98502748
5727734	69359192	65699718	63639618	91677097	34756225	40373192
14265869	11488232	34431957	79388270	74590674	1054181	8961557
42530301	30897482	3275739	78595706	58597066	57197252	51196731
92516182	17285242	4609029	5285224	11455545	6666794	45778060
67007481	77518764	95643553	82793232	30346760	79811270	28902650
44134928	74552913	10557612	51389552	13963047	82539007	40480327
75360636	50345784	44724937	29483861	98443677	69107253	91778450
67411495	34594063	46895605	24899390	23329986	58350973	26586882
71792839	77174523	75251162	57668806	29235769	98527155	1562386
63305774	95334060	81194304	12795899	32189733	80199873	89311937
70885836	63048000	91482317	58199449	58933392	36434030	62965593
54938457	64446279	31679807	53188021	7787762	39296234	50862249
78225149	13071968	2130336	68769200	7606841	49962871	10167035
13028713	59740001	44883538	62007340	75459660	17482431	35949768
67241858	94262831	29722640	64608800	79002077	81978675	66742319

10-130

66017696	16299012	30412116	39335476	81062803	71551611	1918220
7175249	43977300	69401533	48862541	60359030	10888726	33020269
59433032	18717725	24421979	9059306	2392172	52673994	42872523
27668647	36207594	1179537	30548298	29939015	17776768	88694308
10047236	65361812	87243529	18827031	12372840	41888846	61055783
15914965	98170646	38704044	34576163	8789612	37271211	82149653
44 98100	74414164	93258734	37063886	65116746	38235665	11637399
22246990	85598925	39108236	49445142	75326139	19412202	99638301
34677344	36606237	75595177	71703443	4837629	77155898	88170012
24683495	79241234	98974321	98833007	15750524	35576034	85309773
92065540	90664447	1394837	47495708	32666830	35017914	87945053
4447892	35637597	72348343	946835	60924752	87892554	3959181
5193404	70157888	31787966	89566376	30660494	47297651	60829322
87643345	17411801	62276945	9483821	30197989	55386496	39290767
31284223	11303886	35215531	10093604	79283617	83318673	47921246
21065915	47769107	36111458	24677627	48382415	18059818	38456160
40511025	69867570	5916970	24171804	45823811	72549905	34074763
53935657	56384551	5586912	117483	83765669	6191758	79530553
50 15003	96701148	38354542	400866	2734579	21472931	89295632
14852396	2682488	32461935	2811114	27045445	87026741	74833675
3604396	63237581	56000350	69013168	79057808	39171164	22097174
67761337	83960452	65096922	17424662	21424967	82166245	33066529
87307585	56594043	86197244	93595503	47163962	58032974	73336723
22641613	82910279	81814386	8423719	78860882	2243858	41105387
76810103	24061031	16603814	15062017	68145894	24922735	98571808
49765593	97555114	36857524	31174606	14604081	69080960	23201504
73862251	42663553	21400296	59218842	13551069	35640941	32202867
84475540	35436206	44030928	98776356	32676913	66357078	34217422
19250504	20374434	56835222	67241254	77271312	89792907	45377075
90968860	603592	68607752	9930712	60264937	48251538	51703044
80400025	82774232	32716241	31020809	5539075	73036501	32616961
56682458	64704513	64339581	52788217	34929000	2566247	57177053
19836358	86470469	88914009	16584329	3787645	45795852	91323350
15379784	75029304	49001853	87121942	68640736	72583449	38198878
50500056	89684212	74614655	2631748	81456804	15380112	84256487
62390382	53146742	54500757	27347897	5567254	18716383	77478204
28365478	43428101	54563845	57287240	32571206	19102443	49361162
54605929	33220227	36687546	81250480	22277181	31382568	93556868
34039322	73167452	92606567	9827984	17359478	66739406	52318306
83150289	8621249	33444724	67222529	87376997	37123532	3315570
42465981	29221531	32836907	70111806	31575561	19842377	30715673
1228803	74305098	18433559	83425985	12646071	77918385	63354222

WS-174

58188451	17976914	56235948	3612439	21065020	22591276	9024477
67743526	55803816	43113470	61843370	26115057	86130123	84418927
79427372	61685612	17306563	54547927	12140751	89510004	5693631
49509481	6240453	13132372	68161745	68184403	34502786	92056863
30980271	50565891	62683481	37507152	52673916	40678254	6230000
30616176	8518548	62202967	75843947	90693226	5689348	63239334
46996225	36934158	39532154	14127652	15529482	23589717	45316373
24558680	57649731	29495375	19698360	96672551	39179993	2036093
69909379	17623460	49911116	30608275	54685803	86351498	3830681
34276775	42226759	81693465	96376877	42880376	81192216	88274733
17868448	23610597	95834402	82982203	77540833	45392391	15587128
2379001	84805159	5925336	34441197	31075919	56947801	13049291
12657151	44464637	88789288	56148958	88852481	30530367	10488528
49077635	71511320	76425289	70242582	86642127	95346837	57180432
39153731	68548956	89434427	27071243	7460790	92450937	60883650
5367123	33084457	54780971	5702036	17520580	35402424	33211479
4241307	18662923	93363797	350893	73306427	59770346	93232220
27959522	31721129	88379190	3626645	18050117	33986438	61848054
23665998	43743446	67370322	84289013	98991892	56263393	86872869
41127061	15446938	85134225	82868477	11173093	93448973	85782226
95394936	68107592	21165936	42953013	61734590	52961447	76337266
9929080	14354078	77034222	20666044	73025524	72196015	10675496
74305383	26451080	9554306	74541982	25704222	71386538	55762130
78376881	76703303	81856161	74655206	69656637	34589377	59607982
89067054	27026065	78706192	79743279	52281779	15069429	68382055
91853673	36075278	37007951	60404699	87673694	21080810	64136914
82455958	71729877	50779769	39627679	65610713	82884934	59938891
2843686	3825592	75658382	9110445	70246478	62025325	44536551
98583155	10519174	9351780	14295140	29623517	666149	83609186
33251123	45750786	31501770	91463123	55111388	25538908	92035561
52611950	42302628	68530297	64625216	35109612	64427586	89298563
97306275	15926656	24404633	19020078	74395187	2649331	39224769
53699494	35072485	12504824	43910468	68355005	1981065	30278599
41377773	89068394	64722481	86980408	69244361	56539196	61039295
88951160	58870106	924620	39367182	33480938	68375975	55030943
44189355	81321597	76460228	54720400	32253354	28363362	61586578
49888599	68256455	48178822	19347590	26826317	96609086	74858570
19260078	18870580	29518701	4808688	34166698	32586393	2060418
14291082	49682992	55553714	2568425	44906523	37524774	11539016
73142734	68336976	31666381	24622389	4343965	64789177	84201031
39831788	84496698	35934483	78432738	37440381	53343419	84423281

10-131

WS-174

39110337	55763502	42335967	66887423	55531812	41601471	73829981
74379164	88766746	34582090	41308199	10884870	58762338	56240244
94291565	32751227	75710770	89003407	42482856	38368423	61391634
68718232	19610132	30592060	1184213	27874185	46969251	83417283
3411388	2312368	89356831	98898308	1431645	9223815	38306654
66519683	45162317	47148614	28921528	45918243	44723723	35122310
71987219	42478637	53898702	1978498	92282371	16463691	99702062
36078853	3360887	58030702	9421604	99511578	46157936	76797937
84464509	69928759	62184745	86582587	62295597	97351695	85706842
24513329	26189279	26634120	58498645	20094967	94177950	57798796
62190855	3548807	3540062	8344715	3831313	4337235	9680872
11174284	13175527	74223735	11259146	66427132	20116833	57850255
33422646	39138318	37608953	97932167	40781659	96611425	64434956
51490186	26391657	82717477	46716937	98946644	99177867	53556409
36386570	70014561	91440416	87414554	60630190	43135184	1605472
67426679	59091918	66847635	9971375	886335	15629807	28971799
54589410	34701383	41774435	25407765	53059836	25610312	77888958
40830399	42875189	69491053	17071401	15720511	16139633	91025006
78186378	86247884	41787468	71741893	14100570	58974388	74544650
46541209	72551418	76637956	13119347	33519549	80825455	27985180
82434475	4248460	62430094	22851249	74737727	67415507	13240532
23873730	84447909	5596481	873827	20354855	42916475	22974980
65834517	97662101	93788806	80061052	76361668	25309068	95122240
47128488	70205557	16540077	94072485	43683891	2602386	60761539
59843614	6898619	96507501	93670931	21594734	26154592	26753689
66729218	34148468	56617247	61585757	26792511	85114477	93740010
85853089	75622434	33181576	6062066	79430476	53955724	20356479
88602281	44460617	60548401	83922690	10808842	67452573	53879349
77712414	40411559	19851225	68906348	77133020	75944659	25558621
1921380	96071276	35753283	50423843	93170091	24998436	80682522
117921	96087334	84838204	24958716	37035037	58853019	57109955
11306810	35475119	64976168	77346285	75362159	93190294	56471336
31073426	32052080	3337351	67012727	90879694	16851269	21600613
57322860	95096636	91157731	53863811	94178772	80923016	31426164
13956108	21551185	9008731	27435681	59157291	26850079	93984815
71702180	76923708	42910130	91497800	91108668	15817736	47312591
78463053	84945113	34123812	34091819	73599899	44242824	22838141
52856361	72884258	44736812	95111800	21869473	34878660	62813778
79098656	80060215	52815473	94381714	18654898	1824999	69068864
85574197	14557328	23430654	29331104	96103857	12786616	5149893
54221688	95521217	90299566	71453741	67245502	62556851	99074425
96351841	9664919	99260915	39993797	78675127	11139806	86749428

87816668	16644111	74941466	69560317	86534113	40819066	75190393
92966770	76604657	15887486	51052469	60719309	79803405	76082645
94776443	98151939	49316099	69011730	88912997	82139876	38190372
79647661	87059625	60390899	80517012	18368873	8532259	45262895
21356832	44464172	75708068	26288556	888076	18549613	8273098
2392548	63251496	44799136	77766898	96741604	10677339	14125707
95031142	34527877	2620165	24043308	54897417	45978467	73385010
38857524	57255168	69856885	2628463	50319380	72621993	48763538
93813148	56296042	65819154	68128680	77535350	25364259	37123636
1152627	78831849	24070938	26868657	79743777	9261099	13425429
31515722	81302780	31085687	63302344	5696592	4656409	36702803
58638152	52343551	7987514	21505867	76315702	74386727	33090992
93770979	17538195	76633958	34866721	60395009	83513420	23549677
84265488	42744331	12117691	39604074	33534190	90047821	55593270
25686216	65548522	55154625	33907693	89029814	53145083	7830342
29182522	76995966	18203301	48338473	81409306	21360076	1505530
74465173	92423096	82976630	89362231	9344249	70871107	51350048
65013611	89667515	7553607	28797394	28817108	73437159	29500156
19978071	46559754	10870277	50890291	61786631	40565860	65626319
12070244	52412981	82120080	18167055	2080978	22424287	8968137
93418859	43931956	1759353	42629473	2197613	82334314	5126122
56452115	27225811	60747036	54503415	59908270	80948395	74290778
18241453	47875524	77007697	79780282	96683043	65901853	1256418
45854364	58204262	60113316	77869278	24070355	10467903	5733177
88262058	63952914	46841813	17209572	10385205	95220734	72365630
84605559	11366757	59050304	96589649	67704591	92588003	24462866
46597966	89947277	93093253	10902031	4658745	61701269	55110690
5903014	34219621	45002926	65146083	58227639	46658359	61663335
61706340	63548581	80792516	10198989	91147989	48240691	14994446
91421832	46337271	21514496	50628291	73268245	79000168	28275347
58030800	12178507	77906920	9260262	49728996	7545058	56735337
75883252	61800842	37691615	7431870	93795123	25660216	41384511
10184025	72839098	90745500	31363165	84589291	90591917	14976882
15018677	15278973	57621391	37550661	68694511	23387800	2201708
63315710	29937958	28151180	54668609	51681855	49648585	27151981
73528596	51637398	6959182	83854789	97365685	76615502	52539821
26460455	4853110	26583408	57948290	57072390	28283040	6012160
17669417	61683887	32689047	71314970	55701816	51540880	39262971
54643128	5947811	86473787	88579353	48830601	49278176	75916828
73219760	87722132	41544600	84567405	11773443	91293383	36790908
36204516	48808033	3353826	67124899	25238148	69962789	230401

10-133

WS-174

7619424	1290293	59290057	3680910	68219329	77504848	62154511
54873787	52316116	41501947	92627507	62033913	17695742	80565629
43868230	88089743	48285112	56671109	13873824	64298138	94887620
23548429	91776182	3608903	55809218	60862938	93792940	98466543
41277020	73292117	19286526	53825443	88565284	4267328	56347235
29567924	47355828	86831838	94816706	41886216	31114204	60226362
36814798	37181097	7012878	6257195	65805200	78232760	48872994
85983757	32357309	72880331	68481533	71762092	17516916	66545771
91529307	37928604	97655819	51283482	74494499	2648024	2456697
70452701	92057165	91622382	3488	98123210	46428171	28545040
36545407	28445827	22077016	37554562	44218366	5882780	1874640
7605132	1885874	36752855	39452499	17670183	83232734	46234111
72243055	41905257	29898121	75320450	7430534	56211233	95920789
97272319	212299	66742043	47035079	56219630	63706834	56512739
32084527	40158988	28150752	42628261	29658051	32640603	50682389
61152829	61624927	54967586	6682597	87689409	60516573	13648915
37439042	15675134	62214265	83392570	42379396	80657486	77820767
1552953	66375335	10946123	37080095	92392497	27482509	34329118
38326932	28992259	93292145	6425636	82956174	50772023	90155340
40669609	31467305	61429754	67355070	57878205	14396015	48831856
2445586	78903169	71803185	99789162	70426838	35576830	20474252
46308403	14716268	65490764	90997958	6210247	87257574	73699684
32787660	26889507	82540966	47226442	70868271	71109553	48344089
28018816	89146703	93097390	11944814	80404904	85810845	95188382
28764174	29247786	33932060	8499964	76275215	9162263	51577533
67688161	33038107	93056075	38765212	10502729	45922073	18250231
26379730	15626692	41637947	26218379	39958892	36263830	25280334
92093653	76223276	24427279	67256405	96955538	81226495	43928854
48712689	5523939	20610859	88920426	62887426	3455084	23593582
19827488	71803313	33965837	19178756	2464165	64690262	8106005
13438682	95234181	43120919	2960607	71781272	20211133	26310696
86851990	30906854	25957652	87252102	76368030	70064925	54912756
74836728	75897125	9731548	94596717	27247207	45617719	46203311
11094471	49605178	14000814	71245370	33260388	37955107	37922684
65334153	66317303	7512046	67577262	273521	44187718	83227935
45448030	97798362	34722040	86019106	48334806	3651283	40069176
51727448	78011101	20736121	31141542	23983159	55098662	70188660
46122072	7408383	96273780	95381229	85160191	44825995	53840293
46081864	84250379	85895384	57491201	81976031	47805914	96736146
83382084	86947112	53543375	72371771	93879183	54485879	11458329
84985319	91443281	33794015	7379859	75714741	79793357	61851239
11878949	3556964	71074234	10239788	30931801	29201887	84894540

PX 71900-10-(174)

78009640	80371503	85709556	64333538	17006358	65001512	17738086
77940365	10599344	79507293	67461364	95314251	80006441	16458903
99225429	49376871	33798161	64870563	92022471	87021780	69490266
61082714	15183908	67640235	26977723	60943040	65493336	29134845
7721098	42975824	81947795	3343203	97421440	34483151	76562742
44708748	50013174	83096361	50643324	93518220	20548971	793832
51664545	99560561	81953478	28997666	65524261	93799886	25433558
76022956	26238866	79421182	60935585	66964909	87857529	92389813
86194876	26979746	98360155	1196599	81521167	81187313	84562960
20945968	28904769	47972815	81095654	56180490	36375709	32912119
38466141	52857610	40440609	31769415	46833954	35385235	86527385
62140416	61011304	11964429	95336391	31752627	34945941	76491418
51512557	61367486	60483684	27951033	61346972	20261840	72160657
89738287	5919459	94191492	36104474	53033362	86159265	98925228
59942983	97488189	33194752	70554038	4127660	19569455	94240310
67258294	19763789	58250368	14974089	70664694	15973194	6853901
74916109	27314599	8363525	70246427	60590610	83166284	55933875
86831841	94901101	138076	32110702	41858453	55401453	23051148
49885620	50644110	88108771	46703318	75642894	71884311	84030453
73028669	80725300	72728650	96411374	73813958	26278852	35207162
43095246	86043014	49817757	88619721	59275185	17713496	93739525
72209237	42820174	29676535	15755189	32592473	38883148	56237308
24147306	27743184	36659202	89193239	87157934	7413413	70641928
46616592	94648082	80920872	68044119	92895512	85265155	44846941
48937986	35268041	10891764	89512868	86262689	55623426	68718015
13505563	97659160	11053611	76658975	67547676	72504053	86359430
92753235	78516987	90887219	94324240	12428935	9321258	95184883
64549244	45375220	69409404	1851336	4665875	64514388	4342374
20031715	79259209	67769895	50649363	46131267	98794941	18632796
51147816	58633592	58280936	69596130	20271224	33147260	76702792
92232821	98935229	72852202	82472599	2969693	58950027	94297842
75115893	10424445	20495159	97585987	31232334	27975484	78105411
90213607	69203974	23945508	66742153	52433298	50182001	640953
45892664	7418668	851085	51673413	80596615	18748166	75471300
76499431	8496047	28157213	90169151	26542702	36895904	37125674
58484938	37147566	3254474	17246990	58484898	36022299	23016046
13129480	2685844	86648035	24407915	34804570	12874996	75506401
3819894	85755058	2202792	93810423	73421343	14101350	80917079
96945130	56867693	41180104	31239269	88850779	27489640	33823961
54504010	67551597	83580981	45104181	95044451	6278880	4751483
56249050	86033735	57219592	8569248	12082832	70765455	31974525

WS-174

THE RAMO-WOOLDRIDGE CORPORATION
Los Angeles 45, California

Evaluation of $|A - \lambda I|$ for Matrix A
Complex Single Precision Floating Point

	Specifications	
Identification Tag:	DET-2	
Type:	Subroutine available on cards for assembly	
Regional Addresses Used:	OOR, O1R, O2R, O3R, CON, FOO, RWT, COO	
Storage:	132 words total program storage Temporary storage used but not stored with program: 4n cells directly addressable by SNIP n(n+1) cells all in ES or all in MD where n is the order of the matrix. The constant and temporary storage pools are used.	
Entrance and Exit:	RJ OOR01 OOR02	
Mode of Operation:	Floating Point. SNIP must be activated in ES.	
Machine Time:	Approximately $n^2(5n+22)$ms (see text)	
Coded by:	W. L. Frank	July, 1956
Approved by:	W. F. Bauer	July, 1956

Description

This subroutine employs complex floating point arithmetic (SNIP) in order to evaluate the determinant of the matrix $A - \lambda I$, where A is a square matrix of order n. For the special case $\lambda = 0$ one obtains the determinant of A. Furthermore, one has obtained an eigenvalue λ if the quantity

$$|A - \lambda I| = 0$$

The following options are provided:

Option No. 1 - A must be stored row by row in the memory (ES or MD). The location of the first element is supplied in the parameter words.

Option No. 2 - A must be supplied to this subroutine row by row by an auxiliary routine to be prepared by the programmer to fit his needs. This auxiliary must have the form which allows for entry from this subroutine by an RJ 00X00 00X01 instruction assuming it is stored starting at location 00X00. The auxiliary will be entered n times, each time requiring the succeeding row of A to be set up in a region TEM00 preassigned by the programmer in the parameter words.

In addition to the 132 words of storage needed by the subroutine, it is necessary to provide 4n cells of temporary storage addressable by SNIP and a block of n(n+1) cells either all on ES or all on MD.

Operating Instructions

1. SNIP must be activated.
2. Entrance to the subroutine is made by the following instructions assuming λ is in the floating accumulator.

   a. Option No. 1 -

P	RJ	00R01	00R02
P+1	00	MAT00	vvvvv
P+2	00	TRI00	TEM00

where OOROO is the location of the first word of the subroutine.

MATOO is the region in which the matrix A is stored row by row.

vvvvv is the order of the matrix A.

TRIOO is the location of the first cell of the block of n(n+l) cells either all in ES or all in MD.

TEMOO is the location of the first cell of the block of 4n cells of temporary storage directly addressable by SNIP.

b. Option No. 2 -

P	RJ	OOR01	OOR02	
P+l	40	OOXOO	vvvvv	BRR
P+2	00	TRIOO	TEMOO	

where OOROO, vvvvv, TRIOO and TEMOO are as above.

OOXOO is the location of the first word of the auxiliary which supplies the successive rows of A. The auxiliary must provide successive rows of the matrix A and place each row in the first 2n cells of TEMOO.

3. Upon exit from the subroutine control is transferred to the word in cell P+3. The value of $|A - \lambda I|$ is left in the floating accumulator and also in the ninth and tenth words of the R-W temporary pool. The input λ can be found in the fifth and sixth words of the R-W temporary pool.

Machine Time

Computing time for operation of this subroutine is given by $n^2(5n + 22)$ ms where n is the order of the matrix.

In case the block of n(n+l) words is stored on MD the time must be increased by $8.5(n^2 + 3n)$ milliseconds. These times are approximate and will be conservative in most cases.

Mathematical Method

Elementary row operations are performed on the matrix $A - \lambda I$ reducing it to an upper triangular matrix $\bar{A}$. Before eliminating, leading elements of two rows which are to be linearly combined are compared and the element of largest magnitude becomes the pivotal point. The product of the diagonal elements of $\bar{A}$ is the value of $|A - \lambda I|$.

```
D              00R00 00500                                 00764  00 00000 00000
D              01R00 00546                                 01042  00 00000 00000
D              02R00 00562                                 01062  00 00000 00000
D              03R00 00600                                 01130  00 00000 00000
D              CON00 00625                                 01161  00 00000 00000
D              F0000 00002                                 00002  00 00000 00000
D              C0000 00003                                 00003  00 00000 00000
D              RWT00 00023                                 00027  00 00000 00000
00R00    MJ 00000 00000                                    00764  45 00000 00000
00R01    MJ 00000 00000           EXIT                     00765  45 00000 00000
00R02    54 00R01 A0015 BRR                                00766  54 00765 20017
00R03    TU A0000 00R06                                    00767  15 20000 00772
00R04    AT 00015 A0000                                    00770  35 00017 20000
00R05    TU A0000 00R14                                    00771  15 20000 01002
00R06    TP 00000 A0000                                    00772  11 00000 20000
00R07    TV A0000 CON06           N                        00773  16 20000 01167
00R08    TU A0000 03R01                                    00774  15 20000 01131
00R09    TP CON01 Q0000                                    00775  11 01162 10000
00R10    LA A0000 00016                                    00776  54 20000 00020
00R11    TU A0000 CON04                                    00777  15 20000 01165
00R12    QS A0000 01R04                                    01000  53 20000 01046
00R13    TU 01R04 03R00           S                        01001  15 01046 01130
00R14    TP 00000 A0000            E                       01002  11 00000 20000
00R15    TV A0000 02R01             T                      01003  16 20000 01063
00R16    TV A0000 03R01                                    01004  16 20000 01131
00R17    TU 00R14 02R29           A                        01005  15 01002 01117
00R18    LA A0000 00015             D                      01006  54 20000 00017
00R19    TU A0000 01R05              D                     01007  15 20000 01047
00R20    TU A0000 02R27               R                    01010  15 20000 01115
00R21    LA A0000 00042                E                   01011  54 20000 00052
00R22    TV A0000 01R05                 S                  01012  16 20000 01047
00R23    QS 01R05 02R06                 S                  01013  53 01047 01070
00R24    QS 01R05 02R12                   E                01014  53 01047 01076
00R25    QS 01R05 02R04                    S               01015  53 01047 01066
00R26    RA 02R04 CON04                                    01016  21 01066 01165
00R27    TP CON02 Q0000                                    01017  11 01163 10000

00R28    RA 02R01 CON06                                    01020  21 01063 01167
00R29    RA 02R01 CON06                                    01021  21 01063 01167
00R30    QS 02R01 02R06                                    01022  53 01063 01070
00R31    RS CON06 00016                                    01023  23 01167 00020
00R32    ST 00016 RWT02                                    01024  36 00020 00031
00R33    TN F0000 RWT04                                    01025  13 00002 00033
00R34    TN C0000 RWT05                                    01026  13 00003 00034
00R35    QS 03R01 03R04                                    01027  53 01131 01134
00R36    TU 00R06 00R37                                    01030  15 00772 01031
00R37    TP 00000 A0000                                    01031  11 00000 20000
00R38    SJ 00R39 00R45                                    01032  46 01033 01041
00R39    TP 01R07 03R02                                    01033  11 01051 01132
```

PX 71900-10-(175)

```
00R40    TU 03R01 03R02                               01034  15 01131 01132
00R41    SP 03R02 00057                               01035  31 01132 00071
00R42    TV A0000 03R02                               01036  16 20000 01132
00R43    RA 03R02 00016                               01037  21 01132 00020
00R44    MJ 00000 01R00                               01040  45 00000 01042
00R45    TP 03R24 03R02                               01041  11 01160 01132
01R00    TP CON00 RWT08                               01042  11 01161 00037
01R01    TP 00013 RWT09                               01043  11 00015 00040
01R02    TU 00013 03R06                               01044  15 00015 01136
01R03    RJ 03R06 03R00                               01045  37 01136 01130
01R04    RP 30000 01R06                               01046  75 30000 01050
01R05    TP 00000 00000                               01047  11 00000 00000
01R06    TN 00016 RWT03                               01050  13 00020 00032
01R07    RJ 03R06 03R00         GET ITH ROW           01051  37 01136 01130
01R08    TU 01R04 02R00                               01052  15 01046 01062
01R09    54 01R05 A0015 BRR                           01053  54 01047 20017
01R10    TU A0000 02R01                               01054  15 20000 01063
01R11    RA RWT03 00016                               01055  21 00032 00020
01R12    TP RWT03 RWT01                               01056  11 00032 00030
01R13    TP 00013 RWT00                               01057  11 00015 00027
01R14    TU 02R06 02R02                               01060  15 01070 01064
01R15    TP RWT00 00004                               01061  11 00027 00004
02R00    RP 30000 02R02                               01062  75 30000 01064
02R01    TP 00000 00000                               01063  11 00000 00000
02R02  LDPM 00000 RWT06  B    S                       01064  14 32000 25035
02R03    ZJ 02R04 02R16         SKIP IF ZERO          01065  47 01066 01102
02R04  LDPM 00000 00000                               01066  14 30000 24000
02R05    TJ RWT06 03R07         INTERCHANGE           01067  42 00035 01137
02R06  LDDV 00000 00000  B                            01070  14 32000 20000
02R07    TN F0000 RWT06                               01071  13 00002 00035
02R08    TN C0000 RWT07                               01072  13 00003 00036
02R09    TP CON02 Q0000         L                     01073  11 01163 10000
02R10    QS 02R01 02R11          I                    01074  53 01063 01075
02R11  LDMP RWT06 00000           N                   01075  14 30035 14000
02R12  ADNO 00000 00000  BS        E                  01076  14 07000 00000
02R13    RA 02R11 CON03             A                 01077  21 01075 01164
02R14    RA 00004 CON05              R                01100  21 00004 01166
02R15    TJ CON04 02R11               L               01101  42 01165 01075
02R16    TP 00021 Q0000                Y              01102  11 00025 10000
02R17    TP 02R01 A0000                               01103  11 01063 20000
02R18    QA 02R00 02R01          COMBINE              01104  52 01062 01063
02R19    RS 02R00 CON05                               01105  23 01062 01166
02R20    RA RWT00 CON05                               01106  21 00027 01166
02R21    IJ RWT01 01R15         R                     01107  41 00030 01061
02R22    RA 02R27 CON05           O                   01110  21 01115 01166
02R23    55 02R01 Q0021 BRR                           01111  55 01063 10025
02R24    TV Q0000 02R27             S                 01112  16 10000 01115
02R25    TU 02R00 02R26                               01113  15 01062 01114
02R26    RP 30000 02R28                               01114  75 30000 01116
02R27    TP 00000 00000                               01115  11 00000 00000
02R28    IJ RWT02 01R07                               01116  41 00031 01051
```

```
02R29    TU 00000 00004                              01117  15 00000 00004
02R30  LDMP 00000 RWT08  B     S   PRODUCT           01120  14 32000 15037
02R31    RA 00004 CON04                OF            01121  21 00004 01165
02R32    RS CON04 CON05            DIAGONAL          01122  23 01165 01166
02R33    IJ CON06 02R30                              01123  41 01167 01120
02R34    RA 00R01 CON03            ELEMENTS          01124  21 00765 01164
02R35    TN RWT04 RWT04                              01125  13 00033 00033
02R36    TN RWT05 RWT05                              01126  13 00034 00034
02R37    MJ 00000 00R01                              01127  45 00000 00765
03R00    RP 30000 03R02                              01130  75 30000 01132
03R01    TP 00000 00000            NEXT              01131  11 00000 00000
03R02    RA 03R01 CON04             ROW              01132  21 01131 01165
03R03    TU 03R06 00004                              01133  15 01136 00004
03R04  LDAD RWT04 00000       BS                     01134  14 30033 07000
03R05    RA 03R06 CON05                              01135  21 01136 01166
03R06    MJ 00000 00000                              01136  45 00000 00000
03R07    TU 02R00 03R19          D                   01137  15 01062 01153
03R08    TU 02R00 03R22           E                  01140  15 01062 01156
03R09    TP CON01 Q0000            T                 01141  11 01162 10000
03R10    QS 02R04 03R23             E                01142  53 01066 01157
03R11    TP 02R02 A0000              R               01143  11 01064 20000
03R12    AT RWT00 A0000               M              01144  35 00027 20000
03R13    QS A0000 03R20                I             01145  53 20000 01154
03R14    TP CON02 Q0000                 N            01146  11 01163 10000
03R15    LA A0000 00057                  E           01147  54 20000 00071
03R16    QS A0000 03R23          PIVOT               01150  53 20000 01157
03R17    54 02R01 A0057 BRR                          01151  54 01063 20071
03R18    TV A0000 03R20                              01152  16 20000 01154
03R19    RP 30000 03R21                              01153  75 30000 01155
03R20    TP 00000 00000                              01154  11 00000 00000
03R21    TN RWT08 RWT08                              01155  13 00037 00037
03R22    RP 30000 01R15                              01156  75 30000 01061
03R23    TP 00000 00000                              01157  11 00000 00000
03R24    RA 03R01 CON04                              01160  21 01131 01165
CON00    01 00000 00000      F   C                   01161  20 14000 00000
CON01    00 00777 00000 B         O                  01162  00 00777 00000
CON02    00 00000 00777 B          N                 01163  00 00000 00777
CON03    00 00000 00002             S                01164  00 00000 00002
CON04    00 00000 00000              T               01165  00 00000 00000
CON05    00 00002 00000               A              01166  00 00002 00000
CON06    00 00000 00000                NTS           01167  00 00000 00000
START          00000                                 00000  45 00000 00000
```

PX 71900-10-(175)

THE RAMO-WOOLDRIDGE CORPORATION
Los Angeles 45, California

Floating Point Linear Matrix Equation Solver (AX=B)

Specifications

Identification Tag:	MTI-1
Type:	Subroutine available on cards for assembly
Regional Addresses Used:	OOM, O1M, O2M, O3M, CON, AUG, RWT
Storage:	171 words total program storage.

Temporary storage used but not stored with program:

$2(n + m)$ cells directly addressable by SNAP and

$$\frac{n(n + 1)}{2} + nm \text{ cells all in ES or all in MD,}$$

where x is an n by m matrix.

The constant and temporary storage pools are used.

Entrance and Exit:	RJ OOMO1 OOMO2
Alarm:	The alarm exit is used to print "SINGUL" if a singularity is detected.
Mode of Operation:	Floating point. SNAP must be in E.S.
Machine Time:	Approximately

$$2n^2(n + 3m + 2) + 3000 \text{ milliseconds}$$

where X is an n by m matrix.

Coded by:	W. L. Frank	June, 1956
Approved by:	W. F. Bauer	July, 1956

Description

This subroutine employs floating point arithmetic in order to solve the linear matrix equation AX = B, where A is a non-singular matrix of size nxn and B has the dimensions nxm. The solution $X = A^{-1}B$, is a matrix of size nxm. For the special case, when B is the identity matrix (I), one obtains the inverse of the matrix A. Otherwise, one can solve m sets of n simultaneous linear equations in n unknowns.

Considerable flexibility is afforded the programmer with respect to the storage of the matrices A, B and the answer X. The programmer must code two auxiliary routines as follows:

(a) The first must provide successive rows of the augmented matrix $\left[A, B\right]$. (When B = I, one only need supply rows of A). Each row, consisting of (n + m) elements (or n elements when B = I), must be transferred to the first (n + m) cells of the 2(n + m) cells of temporary storage provided by the programmer.

(b) The second auxiliary must take the successive columns of X, found in the first n cells of the 2(n + m) cells of temporary storage and either store them internally or punch them out.

These auxiliary routines are automatically entered n and m times respectively by RJ instructions in the subroutine. The subroutine sets up these two RJ instructions from information gleaned from the parameters of the entry. This procedure allows storage of A, B and X on ES, MD, magnetic tape or externally on cards or tape. It is also possible to generate the elements of successive rows when a functional relation exists.

In addition to the 171 words of storage needed by the subroutine, it is necessary to provide 2(n + m) cells temporary storage addressable by SNAP and a block of $\frac{n(n+1)}{2}$ + nm cells, either all on ES or all on MD.

Operating Instructions

1. SNAP must be activated.
2. Entrance to the subroutine is made by the following orders ($B \neq I$):

P	RJ	00M01	00M02
P + 1	00	00X00	00Y01
P + 2	--	uuuuu	vvvvv
P + 3	--	zzzzz	xxxxx

PX 71900-10-(176)

where 00M00 is the location of the first word of the subroutine
00X00 is the location of the first word of the first auxiliary
00Y01 is the location of the second word of the second auxiliary
uuuuu = m (number of columns of B)
vvvvv = n (number of rows of A)
zzzzz = is the location of the first cell of the 2(n + m) cells of temporary storage addressable directly by SNAP.
xxxxx = is the location of the first cell of the block of $\frac{n(n+1)}{2} + nm$ cells all in ES or all in MD.

3. For the case when B = I, the P + 1 word must be 40 00X00 00Y01 BRR.
4. The auxiliary routines must be available and coded so that they can be entered with

    RJ 00X00 00X01
    and RJ 00Y00 00Y01 respectively.

   This implies that the first and second words of both auxiliaries are exit and entrances respectively.

Alarm Conditions

If a singular matrix is detected in the process of inversion, the alarm routine ALR-1 is entered and "singul-wwwww" is printed where wwwww is the address of the cell from which the subroutine was entered. The routine cannot, however, detect all singularities due to round-off errors (see below).

Starting after an alarm print-out will return control to P + 4 in the main program.

Machine Time

The machine time is as indicated on the first page when all operations are carried on in ES. This time is exclusive of the times taken by the auxiliaries. In case the block of $\frac{n(n+1)}{2}$ + nm words is stored on MD, the time must be increased by $8.5(n^2 + 3n + nm^2 + mn)$ milliseconds. These times are approximate and will be a minimum in most cases.

Mathematical Method (Gauss elimination method)

Elementary row operations are performed on the matrix A reducing it to an upper triangular matrix $\bar{A}$. At the same time, these operations are performed on the matrix B giving a new matrix $\bar{B}$. Before eliminating, leading elements of two rows are compared and the element of largest magnitude becomes the pivotal element.

Next, successive columns of $\bar{B}$ are taken and the equation $\overline{AX} = \bar{B}$ is solved by the back substitution procedure.

Singularities in A are detected if a zero appears on the diagonal of $\bar{A}$. Since round-off errors can prevent this from occurring one must consider the magnitudes of the elements of X as compared to those of A. A large order of difference may indicate poor conditioning of A.

Accuracy

The accuracy in the result is a function of the condition of the matrix A. Six to seven decimal place accuracy was obtained for matrices of order 10 to 16. A test was performed in which the highly ill conditioned Hilbert matrices of order 2 - 10 were inverted. Seven place accuracy was obtained for the matrix of order 2. One more digit was lost for each succeeding higher order matrix.

PX 71900-10-(176)

```
D              00M00  00100                                       00144   00  00000  00000
D              01M00  00157                                       00235   00  00000  00000
D              02M00  00179                                       00263   00  00000  00000
D              03M00  00208                                       00320   00  00000  00000
D              AUG00  00255                                       00377   00  00000  00000
D              CON00  00262                                       00406   00  00000  00000
D              RWT00  00023                                       00027   00  00000  00000
00M00    37    75701  75702  B        ALARM                       00144   37  75701  75702
00M01    MJ    00000  00000           EXIT                        00145   45  00000  00000
00M02    54    00M01  20017  BRB      ENTRY                       00146   54  00145  20017
00M03    TU    A0000  00M09                                       00147   15  20000  00155
00M04    TU    A0000  AUG00                                       00150   15  20000  00377
00M05    AT    00015  A0000                                       00151   35  00017  20000
00M06    TU    A0000  00M26                                       00152   15  20000  00176
00M07    AT    00015  A0000                                       00153   35  00017  20000
00M08    TU    A0000  00M19                                       00154   15  20000  00167
00M09    TP    00000  A0000                                       00155   11  00000  20000
00M10    TU    A0000  01M00           SET                         00156   15  20000  00235
00M11    TV    A0000  03M31                                       00157   16  20000  00357
00M12    AT    00015  A0000            A                          00160   35  00017  20000
00M13    SS    00016  00015             U                         00161   34  00020  00017
00M14    TU    A0000  03M31              X                        00162   15  20000  00357
00M15    LA    A0000  00042               I                       00163   54  20000  00052
00M16    TV    A0000  01M00                L                      00164   16  20000  00235
00M17    TP    01M00  01M05                                       00165   11  00235  00242
00M18    TP    CON01  Q0000                                       00166   11  00407  10000
00M19    TP    00000  A0000           S                           00167   11  00000  20000
00M20    TV    A0000  01M03            E                          00170   16  20000  00240
00M21    TU    A0000  01M03             T                         00171   15  20000  00240
00M22    TU    A0000  02M25                                       00172   15  20000  00314
00M23    QS    01M03  02M05                                       00173   53  00240  00270
00M24    QS    01M03  02M10            A                          00174   53  00240  00275
00M25    TU    02M25  02M03              D                        00175   15  00314  00266
00M26    TP    00000  A0000               D                       00176   11  00000  20000
00M27    TU    A0000  CON04           M    R                      00177   15  20000  00412
00M28    TV    A0000  CON06           N     E                     00200   16  20000  00414
00M29    54    CON06  20017  BRB             S                    00201   54  00414  20017
00M30    QS    A0000  AUG02                   S                   00202   53  20000  00401
00M31    AT    02M25  A0000                    E                  00203   35  00314  20000
00M32    QS    A0000  03M20                     S                 00204   53  20000  00344
00M33    QS    A0000  03M29                                       00205   53  20000  00355
00M34    TP    AUG02  A0000                                       00206   11  00401  20000
00M35    AT    CON04  A0000                                       00207   35  00412  20000
00M36    QS    A0000  CON05                                       00210   53  20000  00413
00M37    QS    A0000  01M02                                       00211   53  20000  00237
00M38    RA    02M03  CON05                                       00212   21  00266  00413
00M39    55    CON04  10025  BRB                                  00213   55  00412  10025
00M40    TV    Q0000  CON07                                       00214   16  10000  00415
00M41    TP    CON03  Q0000                                       00215   11  00411  10000
00M42    54    03M20  20071  BRB                                  00216   54  00344  20071
```

```
00M43   QS A0000 AUG03                             00217  53 20000 00402
00M44   TV AUG03 AUG04                             00220  16 00402 00403
00M45   TV AUG03 02M27                             00221  16 00402 00316
00M46   AT CON07 A0000                             00222  35 00415 20000
00M47   QS A0000 02M01                             00223  53 20000 00264
00M48   QS A0000 02M05                             00224  53 20000 00270
00M49   AT CON06 A0000                             00225  35 00414 20000
00M50   QS A0000 03M16                             00226  53 20000 00340
00M51   QS A0000 03M20                             00227  53 20000 00344
00M52   QS A0000 03M25                             00230  53 20000 00351
00M53   QS A0000 03M28                             00231  53 20000 00354
00M54   RS CON06 00016                             00232  23 00414 00020
00M55   ST 00016 RWT02                             00233  36 00020 00031
00M56   RS CON07 00016                             00234  23 00415 00020
01M00   RJ 00000 00000         TO AUX 1            00235  37 00000 00000
01M01   RJ AUG06 AUG00         AUGMENT             00236  37 00405 00377
01M02   RP 30000 01M04                             00237  75 30000 00241
01M03   TP 00000 00000                             00240  11 00000 00000
01M04   TN 00016 RWT03         SET INDEX           00241  13 00020 00032
01M05   RJ 00000 00000         GET ITH ROW         00242  37 00000 00000
01M06   RJ AUG06 AUG00         AUGMENT             00243  37 00405 00377
01M07   TU 01M02 02M00                             00244  15 00237 00263
01M08   54 01M03 20017 BRB                         00245  54 00240 20017
01M09   TU A0000 02M01                             00246  15 20000 00264
01M10   RA RWT03 00016                             00247  21 00032 00020
01M11   TP RWT03 RWT01                             00250  11 00032 00030
01M12   TP 00013 RWT00                             00251  11 00015 00027
01M13   TP CON01 Q0000                             00252  11 00407 10000
01M14   QS 02M05 01M15                             00253  53 00270 00254
01M15   TM 00000 00002                             00254  12 00000 00002
01M16   TU 01M15 03M44                             00255  15 00254 00374
01M17   55 01M15 20025 BRB                         00256  55 00254 20025
01M18   TV A0000 03M46                             00257  16 20000 00376
01M19   RA 01M15 00015                             00260  21 00254 00017
01M20   TP 00002 A0000                             00261  11 00002 20000
01M21   ZJ 02M00 02M14         SKIP IF ZERO        00262  47 00263 00301
02M00   RP 30000 02M02         TRANSMIT ITH        00263  75 30000 00265
02M01   TP 00000 00000         ROW TO ES           00264  11 00000 00000
02M02   TP RWT00 00004         B BOX               00265  11 00027 00004
02M03   TM 00000 A0000                             00266  12 00000 20000
02M04   TJ 00002 03M38                             00267  42 00002 00366
02M05 LDDV 00000 00000  B                          00270  14 32000 20000
02M06   TP 00002 RWT08                             00271  11 00002 00037
02M07   TP CON03 Q0000         L                   00272  11 00411 10000
02M08   QS 02M01 02M09          I                  00273  53 00264 00274
02M09 LDMP RWT08 00000           N                 00274  14 30037 14000
02M10 SUNO 00000 00000  BS        E                00275  14 13000 00000
02M11   RA 02M09 00016             A               00276  21 00274 00020
02M12   RA 00004 00015              R              00277  21 00004 00017
02M13   TJ CON05 02M09               L             00300  42 00413 00274
```

PX 71900-10-(176)

```
02M14    TP 00021 Q0000                  Y        00301  11 00025 10000
02M15    TP 02M01 A0000                           00302  11 00264 20000
02M16    QA 02M00 02M01             COMBINE       00303  52 00263 00264
02M17    RS 02M00 00015                           00304  23 00263 00017
02M18    RA RWT00 00015               R           00305  21 00027 00017
02M19    IJ RWT01 01M15                O          00306  41 00030 00254
02M20    RA 02M25 00015                 W         00307  21 00314 00017
02M21    55 02M01 10025 BRB              S        00310  55 00264 10025
02M22    TV Q0000 02M25                           00311  16 10000 00314
02M23    TU 02M00 02M24                           00312  15 00263 00313
02M24    RP 30000 02M26            REPLACE        00313  75 30000 00315
02M25    TP 00000 00000             REDUCED ROW   00314  11 00000 00000
02M26    IJ RWT02 01M05            N-2 TIMES      00315  41 00031 00242
02M27    TP CON02 00000                           00316  11 00410 00000
02M28    TP CON06 RWT05                           00317  11 00414 00034
03M00    TV 03M16 03M14                           00320  16 00340 00336
03M01    TU 02M01 03M14                           00321  15 00264 00336
03M02    TP 00021 Q0000                           00322  11 00025 10000
03M03    QS 00013 03M13                           00323  53 00015 00335
03M04    TP 00016 RWT01                           00324  11 00020 00030
03M05    TP 00013 RWT03                           00325  11 00015 00032
03M06    RA 03M14 00015                           00326  21 00336 00017
03M07    MJ 00000 03M09                           00327  45 00000 00331
03M08    RS 03M14 CON04                           00330  23 00336 00412
03M09    RA 03M13 00015             TRANSFER      00331  21 00335 00017
03M10    RA RWT01 00015             ROWS OF       00332  21 00030 00017
03M11    TU 03M14 03M16             UPPER         00333  15 00336 00340
03M12    RS 03M14 RWT01             TRIANGULAR    00334  23 00336 00030
03M13    RP 30000 03M15             MATRIX        00335  75 30000 00337
03M14    TP 00000 00000                           00336  11 00000 00000
03M15    RA 03M16 RWT02                           00337  21 00340 00031
03M16    TN 00000 00000             B             00340  13 00000 00000
03M17    TP 00013 00004              A            00341  11 00015 00004
03M18    TP RWT03 RWT00               C           00342  11 00032 00027
03M19    TP 00013 RWT06                K          00343  11 00015 00035
03M20  LDMP 00000 00000   B    B    S             00344  14 32000 16000
03M21  ADNO RWT06 00000    S         U            00345  14 05035 00000
03M22    RS 00004 00015               B           00346  23 00004 00017
03M23    IJ RWT00 03M20                S          00347  41 00027 00344
03M24    RA RWT03 00016                 T         00350  21 00032 00020
03M25  NOLD 00000 00000        B         I        00351  14 00000 32000
03M26    ZJ 03M27 03M36                   T       00352  47 00353 00364
03M27    TN RWT06 00002                    U      00353  13 00035 00002
03M28  NODV 00000 00000        B             T    00354  14 00000 22000
03M29  STNO 00000 00000   B                   E   00355  14 36000 00000
03M30    IJ RWT05 03M08                           00356  41 00034 00330
03M31    RJ 00000 00000            TO AUX 2       00357  37 00000 00000
03M32    RA RWT02 00015                           00360  21 00031 00017
03M33    IJ CON07 02M27                           00361  41 00415 00316
```

03M34	RA	00M01	CON08				00362	21	00145	00416
03M35	MJ	00000	00M01				00363	45	00000	00145
03M36	11	CON00	75756	BRB		TO ALARM	00364	11	00406	75756
03M37	MJ	00000	00M00				00365	45	00000	00144
03M38	TU	02M03	03M46			INTERCHANGE	00366	15	00266	00376
03M39	TU	02M00	03M43				00367	15	00263	00373
03M40	TU	02M00	03M45			R	00370	15	00263	00375
03M41	55	02M01	20025	BRB		O	00371	55	00264	20025
03M42	TV	A0000	03M44			W	00372	16	20000	00374
03M43	RP	30000	03M45			S	00373	75	30000	00375
03M44	TP	00000	00000				00374	11	00000	00000
03M45	RP	30000	02M00				00375	75	30000	00263
03M46	TP	00000	00000				00376	11	00000	00000
AUG00	TP	00000	A0000			TEST TO SEE	00377	11	00000	20000
AUG01	SJ	AUG02	AUG06			IF INVERT	00400	46	00401	00405
AUG02	RP	10000	AUG04				00401	75	10000	00403
AUG03	TP	00013	00000			AUGMENT	00402	11	00015	00000
AUG04	TP	CON02	00000			IDENTITY	00403	11	00410	00000
AUG05	RA	AUG04	00016				00404	21	00403	00020
AUG06	MJ	00000	00000				00405	45	00000	00000
CON00	24	14061	33411	B			00406	24	14061	33411
CON01	00	00777	00000	B		CONSTANTS	00407	00	00777	00000
CON02	01	00000	00000		F		00410	20	14000	00000
CON03	00	00000	00777	B		AND	00411	00	00000	00777
CON04	00	00000	00000				00412	00	00000	00000
CON05	00	00000	00000			TEMPORARY	00413	00	00000	00000
CON06	00	00000	00000			STORAGE	00414	00	00000	00000
CON07	00	00000	00000				00415	00	00000	00000
CON08	00	00000	00003				00416	00	00000	00003
START		00M00					00000	45	00000	00144

PX 71900-10-(176)

THE RAMO-WOOLDRIDGE CORPORATION
Los Angeles 45, California

Complex Single Precision Floating Point Linear Matrix Equation Solver (AX = B)

Specifications

Identification Tag: MTI-2

Type: Subroutine available on cards for assembly

Regional Addresses Used: OOM, O1M, O2M, O3M, CON, AUG, RWT

Storage: 185 words total program storage.

Temporary storage used but not stored with program:

$4(n + m)$ cells directly addressable by SNIP, and

$n(n + 1) + 2nm$ cells all in ES or all in MD, where X is an n by m matrix.

The constant and temporary storage pools are used.

Entrance and Exit: RJ OOMO1 OOMO2

Alarm: The alarm exit is used to print "SINGUL" if a singularity is detected.

Mode of Operation: Floating point. Complex mode of SNIP must be activated in ES.

Machine Time: See Text.

Coded by:	W. L. Frank	June, 1956
Approved by:	W. F. Bauer	June, 1956

Description

This subroutine employs single precision complex floating point arithmetic (SNIP) in order to solve the linear matrix equation AX = B, where A is a non-singular matrix of size nxn and B has the dimensions nxm. The solution, $X = A^{-1}B$, is a matrix of size nxm. For the special case, when B is the identity matrix (I), one obtains the inverse of the matrix A. Otherwise, one can solve m sets of n simultaneous linear equations in n unknowns.

Considerable flexibility is afforded the programmer with respect to the storage of the matrices A, B and the answer X. The programmer must code two auxiliary routines as follows:

(a) The first must provide successive rows of the augmented matrix [A, B]. (When B = I, one only need supply rows of A). Each row, consisting of 2(n + m) elements (or 2n elements when B = I), must be transferred to the first 2(n + m) cells of the 4(n + m) cells of temporary storage provided by the programmer.

(b) The second auxiliary must take the successive columns of X, found in the first 2n cells of the 4(n + m) cells of temporary storage and either store them internally or punch them out.

These auxiliary routines are automatically entered n and m times respectively by RJ instructions in the subroutine. The subroutine sets up these two RJ instructions from information gleaned from the parameters of the entry. This procedure allows storage of A, B and X on ES, MD, magnetic tape or externally on cards or tape. It is also possible to generate the elements of successive rows when a functional relation exists.

In addition to the 185 words of storage needed by the subroutine, it is necessary to provide 4(n + m) cells temporary storage addressable by SNIP and a block of n(n + 1) + 2nm cells, either all on ES or all on MD.

Operating Instructions

1. Entrance to the subroutine is made as follows (B ≠ I):

P	RJ	00M01	00M02
P + 1	00	00X00	00Y01
P + 2	--	uuuuu	vvvvv
P + 3	--	zzzzz	xxxxx

PX 71900-10-(177)

where 00M00 is the location of the first word of the subroutine

00X00 is the location of the first word of the first auxiliary

00Y01 is the location of the second word of the second auxiliary

uuuuu = m (number of columns of B)

vvvvv = n (number of rows of A)

xxxxx = is the location of the first cell of the block of n(n + 1) + 2nm cells all in ES or all in MD.

zzzzz = is the location of the first cell of the 4(n + m) cells of temporary storage addressable directly by SNIP.

2. For the case when B = I, the P + 1 word must be 40 00X00 00Y01 (BRR)
3. The auxiliary routines must be available and coded so that they can be entered with

   RJ 00X00 00X01

   and

   RJ 00Y00 00Y01 respectively.

   This implies that the first and second words of both auxiliaries are exit and entrances respectively.

Alarm Conditions

If a singular matrix is detected in the process of inversion, the alarm routine ALR-1 is entered and "singul-wwwww" is printed where wwwww is the address of the cell from which the subroutine was entered. The routine cannot, however, detect all singularities due to round-off errors (see below).

Machine Time

Computing time for operation of this subroutine can be estimated from the following table:

Order (n = m)	Time in seconds
3	1.7
5	4.7
8	15.6
10	27.7

This time is exclusive of the times taken by the auxiliaries. In case the block of n(n + 1) + 2nm words is stored on MD, the time must be increased $34(n^2 + nm + 2nm^2)$ milliseconds. These times are approximate and are conservative in most cases.

Mathematical Method (Gauss elimination method)

Elementary row operations are performed on the matrix A reducing it to an upper triangular matrix $\bar{A}$. At the same time, these operations are performed on the matrix B giving a new matrix $\bar{B}$.

Before eliminating, leading elements of two rows are compared and the element of largest magnitude becomes the pivotal point.

Next, successive columns of $\bar{B}$ are taken and the equation $\bar{A}X = \bar{B}$ is solved by the back substitution procedure.

Singularities in A are detected if a zero appears on the diagonal of $\bar{A}$. Since round-off errors can prevent this from occurring one must consider the magnitudes of the elements of X as compared to those of A. A large order of difference may indicate poor conditioning of A.

Accuracy

The accuracy in the result is a function of the condition of the matrix A. Six to seven decimal place accuracy was obtained for matrices of order 10 to 16.

PX 71900-10-(177)

```
D            00M00  00100                              00144  00  00000  00000
D            01M00  00161                              00241  00  00000  00000
D            02M00  00176                              00260  00  00000  00000
D            03M00  00210                              00322  00  00000  00000
D            AUG00  00267                              00413  00  00000  00000
D            CON00  00274                              00422  00  00000  00000
D            RWT00  00023                              00027  00  00000  00000
00M00    37  75701  75702  B         ALARM             00144  37  75701  75702
00M01    MJ  00000  00000            EXIT              00145  45  00000  00000
00M02    54  00M01  A0015  BRR       ENTRANCE          00146  54  00145  20017
00M03    TU  A0000  00M09                              00147  15  20000  00155
00M04    TU  A0000  AUG00                              00150  15  20000  00413
00M05    AT  00015  A0000                              00151  35  00017  20000
00M06    TU  A0000  00M26                              00152  15  20000  00176
00M07    AT  00015  A0000                              00153  35  00017  20000
00M08    TU  A0000  00M19                              00154  15  20000  00167
00M09    TP  00000  A0000                              00155  11  00000  20000
00M10    TU  A0000  01M00            SET               00156  15  20000  00241
00M11    TV  A0000  03M34                              00157  16  20000  00364
00M12    AT  00015  A0000              A               00160  35  00017  20000
00M13    SS  00016  00015               U              00161  34  00020  00017
00M14    TU  A0000  03M34                X             00162  15  20000  00364
00M15    LA  A0000  00042                 I            00163  54  20000  00052
00M16    TV  A0000  01M00                  L           00164  16  20000  00241
00M17    TP  01M00  01M05                              00165  11  00241  00246
00M18    TP  CON01  Q0000                              00166  11  00423  10000
00M19    TP  00000  A0000                              00167  11  00000  20000
00M20    TV  A0000  01M03            S                 00170  16  20000  00244
00M21    TU  A0000  01M03             E                00171  15  20000  00244
00M22    TU  A0000  02M27              T               00172  15  20000  00313
00M23    QS  01M03  02M06                              00173  53  00244  00266
00M24    QS  01M03  02M12             A                00174  53  00244  00274
00M25    QS  02M27  02M04              D               00175  53  00313  00264
00M26    TP  00000  A0000               D              00176  11  00000  20000
00M27    TU  A0000  CON04            M   R             00177  15  20000  00426
00M28    TV  A0000  CON06            N    E            00200  16  20000  00430
00M29    54  CON06  A0016  BRR             S           00201  54  00430  20020
00M30    QS  A0000  AUG02                   S          00202  53  20000  00415
00M31    AT  02M27  A0000                    E         00203  35  00313  20000
00M32    QS  A0000  03M21                     S        00204  53  20000  00347
00M33    QS  A0000  03M32                              00205  53  20000  00362
00M34    54  CON04  A0001  BRR                         00206  54  00426  20001
00M35    AT  AUG02  A0000                              00207  35  00415  20000
00M36    QS  A0000  01M02                              00210  53  20000  00243
00M37    QS  A0000  CON05                              00211  53  20000  00427
00M38    RA  02M04  CON05                              00212  21  00264  00427
00M39    55  CON04  Q0021  BRR                         00213  55  00426  10025
00M40    TV  Q0000  CON07                              00214  16  10000  00431
```

```
00M41    TP CON03 Q0000                                00215  11 00425 10000
00M42    54 03M21 A0057 BRR                            00216  54 00347 20071
00M43    QS A0000 AUG03                                00217  53 20000 00416
00M44    TV AUG03 AUG04                                00220  16 00416 00417
00M45    TV AUG03 02M29                                00221  16 00416 00315
00M46    AT CON07 A0000                                00222  35 00431 20000
00M47    AT CON07 A0000                                00223  35 00431 20000
00M48    QS A0000 02M01                                00224  53 20000 00261
00M49    QS A0000 02M06                                00225  53 20000 00266
00M50    AT CON06 A0000                                00226  35 00430 20000
00M51    AT CON06 A0000                                00227  35 00430 20000
00M52    QS A0000 03M16                                00230  53 20000 00342
00M53    QS A0000 03M21                                00231  53 20000 00347
00M54    QS A0000 03M26                                00232  53 20000 00354
00M55    QS A0000 03M31                                00233  53 20000 00361
00M56    RS CON06 00016                                00234  23 00430 00020
00M57    ST 00016 RWT02                                00235  36 00020 00031
00M58    RS CON07 00016                                00236  23 00431 00020
00M59    TV 02M29 02M30                                00237  16 00315 00316
00M60    RA 02M30 00016                                00240  21 00316 00020
01M00    RJ 00000 00000                                00241  37 00000 00000
01M01    RJ AUG06 AUG00                                00242  37 00421 00413
01M02    RP 30000 01M04                                00243  75 30000 00245
01M03    TP 00000 00000                                00244  11 00000 00000
01M04    TN 00016 RWT03            SET INDEX           00245  13 00020 00032
01M05    RJ 00000 00000            GET ITH ROW         00246  37 00000 00000
01M06    RJ AUG06 AUG00            AUGMENT             00247  37 00421 00413
01M07    TU 01M02 02M00                                00250  15 00243 00260
01M08    54 01M03 A0015 BRR                            00251  54 00244 20017
01M09    TU A0000 02M01                                00252  15 20000 00261
01M10    RA RWT03 00016                                00253  21 00032 00020
01M11    TP RWT03 RWT01                                00254  11 00032 00030
01M12    TP 00013 RWT00                                00255  11 00015 00027
01M13    TU 02M06 02M02                                00256  15 00266 00262
01M14    TP RWT00 00004            B BOX               00257  11 00027 00004
02M00    RP 30000 02M02            TRANSMIT ITH        00260  75 30000 00262
02M01    TP 00000 00000            ROW TO ES           00261  11 00000 00000
02M02  LDPM 00000 RWT08  B     S                       00262  14 32000 25037
02M03    ZJ 02M04 02M16            SKIP IF ZERO        00263  47 00264 00300
02M04  LDPM 00000 00000                                00264  14 30000 24000
02M05    TJ RWT08 03M41                                00265  42 00037 00373
02M06  LDDV 00000 00000  B                             00266  14 32000 20000
02M07    TP 00002 RWT08                                00267  11 00002 00037
02M08    TP 00003 RWT09                                00270  11 00003 00040
02M09    TP CON03 Q0000            L                   00271  11 00425 10000
02M10    QS 02M01 02M11             I                  00272  53 00261 00273
02M11  LDMP RWT08 00000              N                 00273  14 30037 14000
02M12  SUNO 00000 00000  BS           E                00274  14 13000 00000
02M13    RA 02M11 CON09                A               00275  21 00273 00433
```

PX 71900-10-(177)

```
02M14    RA 00004 CON10                  R           00276  21 00004 00434
02M15    TJ CON05 02M11                   L          00277  42 00427 00273
02M16    TP 00021 Q0000                    Y         00300  11 00025 10000
02M17    TP 02M01 A0000                              00301  11 00261 20000
02M18    QA 02M00 02M01              COMBINE         00302  52 00260 00261
02M19    RS 02M00 CON10                              00303  23 00260 00434
02M20    RA RWT00 CON10              R               00304  21 00027 00434
02M21    IJ RWT01 01M14               O              00305  41 00030 00257
02M22    RA 02M27 CON10                W             00306  21 00313 00434
02M23    55 02M01 Q0021 'BRR           S             00307  55 00261 10025
02M24    TV Q0000 02M27                              00310  16 10000 00313
02M25    TU 02M00 02M26                              00311  15 00260 00312
02M26    RP 30000 02M28             REPLACE          00312  75 30000 00314
02M27    TP 00000 00000              REDUCED ROW     00313  11 00000 00000
02M28    IJ RWT02 01M05             N-2 TIMES        00314  41 00031 00246
02M29    TP CON02 00000                              00315  11 00424 00000
02M30    TP 00013 00000                              00316  11 00015 00000
02M31    TP CON06 RWT05                              00317  11 00430 00034
02M32    TP 00013 RWT03                              00320  11 00015 00032
02M33    TV 03M16 03M13                              00321  16 00342 00337
03M00    TU 02M01 03M13                              00322  15 00261 00337
03M01    TP 00021 Q0000                              00323  11 00025 10000
03M02    QS 00013 03M12                              00324  53 00015 00336
03M03    TP CON09 RWT01                              00325  11 00433 00030
03M04    RA 03M13 CON10                              00326  21 00337 00434
03M05    MJ 00000 03M08                              00327  45 00000 00332
03M06    RS 03M13 CON04                              00330  23 00337 00426
03M07    RS 03M13 CON04                              00331  23 00337 00426
03M08    RA 03M12 CON10             TRANSFER         00332  21 00336 00434
03M09    RA RWT01 CON10             ROWS OF          00333  21 00030 00434
03M10    TU 03M13 03M16             UPPER            00334  15 00337 00342
03M11    RS 03M13 RWT01             TRIANGULAR       00335  23 00337 00030
03M12    RP 30000 03M14             MATRIX           00336  75 30000 00340
03M13    TP 00000 00000                              00337  11 00000 00000
03M14    RA 03M16 RWT02                              00340  21 00342 00031
03M15    RP 30002 03M17                              00341  75 30002 00343
03M16    TN 00000 00000             B                00342  13 00000 00000
03M17    TP 00013 00004              A               00343  11 00015 00004
03M18    TP RWT03 RWT00               C              00344  11 00032 00027
03M19    TP 00013 RWT06                K             00345  11 00015 00035
03M20    TP 00013 RWT07             S                00346  11 00015 00036
03M21  LDMP 00000 00000    B   B     U               00347  14 32000 16000
03M22  ADNO RWT06 00000     S         B              00350  14 05035 00000
03M23    RS 00004 CON10                S             00351  23 00004 00434
3M24     IJ RWT00 03M21                 T            00352  41 00027 00347
3M25     RA RWT03 00016                  I           00353  21 00032 00020
3M26   NOLD 00000 00000                   T          00354  14 00000 32000
3M27   PMNO 00000 00000                    U         00355  14 24000 00000
3M28     ZJ 03M29 03M39                     T        00356  47 00357 00371
3M29     TN RWT06 00002                      E       00357  13 00035 00002
```

```
03M30    TN RWT07 00003                          00360  13 00036 00003
03M31  LDDV 00002 00000        B                 00361  14 30002 22000
03M32  STNO 00000 00000    B                     00362  14 36000 00000
03M33    IJ RWT05 03M06                          00363  41 00034 00330
03M34    RJ 00000 00000          TO AUX 2        00364  37 00000 00000
03M35    RA RWT02 CON10                          00365  21 00031 00434
03M36    IJ CON07 02M29                          00366  41 00431 00315
03M37    RA 00M01 CON08                          00367  21 00145 00432
03M38    MJ 00000 00M01                          00370  45 00000 00145
03M39    11 CON00 75756 BRB      TO ALARM        00371  11 00422 75756
03M40    MJ 00000 00M00                          00372  45 00000 00144
03M41    TU 02M00 03M53                          00373  15 00260 00407
03M42    TU 02M00 03M55                          00374  15 00260 00411
03M43    TP CON01 Q0000                          00375  11 00423 10000
03M44    QS 02M04 03M56                          00376  53 00264 00412
03M45    TP 02M02 A0000                          00377  11 00262 20000
03M46    AT RWT00 A0000                          00400  35 00027 20000
03M47    QS A0000 03M54                          00401  53 20000 00410
03M48    TP CON03 Q0000                          00402  11 00425 10000
03M49    LA A0000 00057                          00403  54 20000 00071
03M50    QS A0000 03M56                          00404  53 20000 00412
03M51    54 02M01 A0057 BRR                      00405  54 00261 20071
03M52    TV A0000 03M54                          00406  16 20000 00410
03M53    RP 30000 03M54                          00407  75 30000 00410
03M54    TP 00000 00000                          00410  11 00000 00000
03M55    RP 30000 01M14                          00411  75 30000 00257
03M56    TP 00000 00000                          00412  11 00000 00000
AUG00    TP 00000 A0000          TEST TO SEE     00413  11 00000 20000
AUG01    SJ AUG02 AUG06           IF INVERT      00414  46 00415 00421
AUG02    RP 10000 AUG04                          00415  75 10000 00417
AUG03    TP 00013 00000          AUGMENT         00416  11 00015 00000
AUG04    TP CON02 00000           IDENTITY       00417  11 00424 00000
AUG05    RA AUG04 CON09                          00420  21 00417 00433
AUG06    MJ 00000 00000                          00421  45 00000 00000
CON00    24 14061 33411 B                        00422  24 14061 33411
CON01    00 00777 00000 B        CONSTANTS       00423  00 00777 00000
CON02    01 00000 00000       F                  00424  20 14000 00000
CON03    00 00000 00777 B          AND           00425  00 00000 00777
CON04    00 00000 00000                          00426  00 00000 00000
CON05    00 00000 00000                          00427  00 00000 00000
CON06    00 00000 00000          TEMPORARY       00430  00 00000 00000
CON07    00 00000 00000                          00431  00 00000 00000
CON08    00 00000 00003          STORAGE         00432  00 00000 00003
CON09    00 00000 00002                          00433  00 00000 00002
CON10    00 00002 00000                          00434  00 00002 00000
START       00M00                                00000  45 00000 00144
```

PX 71900-10-(177)

THE RAMO-WOOLDRIDGE CORPORATION
Los Angeles 45, California

Complex Gill Method Routine

Specifications

Identification Tag:	NUI-5
Type:	Subroutine
Assembly Routine Spec:	SUB 51071 12431
Storage:	124 words total program storage
	The temporary pool is used. In addition, cells 33 through 38 (41b - 46b) are used as temporary storage.
	The constant pool is used.
Entrance and Exit:	RJ GIM01 GIM02 set up RJ GIM01 GIM03 to get next point MJ 00000 GIM04 from derivative calculation

Coded by:	W. Mancina	May, 1956
Approved by:	W. F. Bauer	June, 1956

Description

The Gill Method Subroutine (NUI-5) integrates a system of first order, differential equations using a step-by-step process. Using the values of the variables at a point and the coding for computing the derivative of each of the dependent variables at that point, this subroutine produces the coordinates for the next point of the solution each time it is entered.

A special entrance sets up the subroutine for a particular system of equations, thus allowing the subroutine to solve concurrently several different systems in the same program.

The independent variable is incremented within the subroutine itself.

Notation

The system of equations to be solved is

$$\frac{dy_i}{dx} = f_i(x, y_1, y_2, \ . \ . \ . \ , y_n), \ (i = 1, 2, \ . \ . \ . \ , n).$$

n is the number of equations in the system

q_i are intermediate values of the calculation (zero initially)

Δx is the increment of the independent variable x

Programming and Operating Instructions:

1. Assign NUI-5 to some arbitrary region, say GIM00. (The Complex Arithmetic Interpretive Routine, SNIP, must be in ES during the execution of NUI-5. SNIP begins in register 634 decimal and extends through register 1023 decimal). SNIP complex must be activated from the program with a 37 01541 01713 command before entering NUI-5.
2. In order to solve a given system, the following array of variables, derivatives intermediate values, and parameters should be assigned a region, say 00N00:

00N00	n	Fixed point form scaled 2^0.	
00N01	Δx	(real)	Floating point form
00N02	Δx	(imag.)	"
00N03	x	(real)	"
00N04	x	(imag.)	"
00N05	$\frac{dy_1}{dx}$ }	(real)	"
00N06		(imag.)	"

PX 71900-10-(178)

	00N07	y_1	(real)	Floating point form
	00N08		(imag.)	"
initially zero	00N09	q_1	(real)	"
	00N10		(imag.)	"
	00N11	$\frac{dy_2}{dx}$	(real)	"
	00N12		(imag.)	"
	00N13	y_2	(real)	"
	00N14		(imag.)	"
initially zero	00N15	q_2	(real)	"
	00N16		(imag.)	"
	.		.	
	.		.	
	.		.	

3. In addition, the coding for computing $\frac{dy_i}{dx}$ for all values of i, (i = 1, 2, 3, ... n) should be assigned a region, say DEV00. This coding will use the values in region 00N00 to compute all $\frac{dy_i}{dx}$ as specified by the equations in the system and should store the results in the appropriate cells in region 00N00. It should then exit to NUI-5 with a MJ 0000 GIM04 (see below).

Entrances

Assuming NUI-5 is in region GIM00; the three entrances are GIM02, GIM03, and GIM04. The exit is GIM01.

1. The first entrance, GIM02, is used for setting up NUI-5 only for the particular system to be solved. It is entered by an RJ command followed by a parameter word which specifies the locations of the variables, and the location of the coding for calculating the derivatives:

   RJ GIM01 GIM02
   00 00N00 DEV00

2. The second entrance, GIM03, is the entrance for producing a point of the solution. It is entered by an RJ command: RJ GIM01 GIM03. The use of this entrance command results in four passes through both NUI-5 and the coding for computing the derivatives, and leaves in region 00N00 the new values of q, the variables, the derivatives at those values, and "x" advanced by "Δx", ready for the next step.

3. The third entrance, GIM04, is the entrance from the coding for calculating the derivatives and is used on each of the four passes necessary for computing one point. As noted above, it is entered by an MJ command in the DEV00 region:

MJ 00000 GIM04

Mathematical Analysis

Theory. "A Process for the Step-by-Step Integration of Differential Equations in an Automatic Digital Computing Machine" by S. Gill, published in Cambridge Philosophical Society Proceedings, Vol. 47, Part I, January 1951, should be consulted for a detailed analysis of the process on which the subroutine is based.

Suppose we know the point $(X, Y_1, Y_2, \ldots, Y_n)$ on the curve defined by the system of equations

$$\frac{dy_1}{dx} = f_1 (x, y_1, y_2, \ldots, y_n)$$

$$\frac{dy_2}{dx} = f_2 (x, y_1, y_2, \ldots, y_n)$$

$$\vdots$$

$$\frac{dy_n}{dx} = f_n (x, y_1, y_2, \ldots, y_n)$$

The Gill Method is a process by which we can find the next point on the curve: i. e. the value of $y_1, y_2, \ldots, y_n$ for $x = X + h$.

The process can be better understood if the case where n=1 is first considered.

We have the point (X,Y) on the curve $\frac{dy}{dx} = f(x,y)$, and we want to find y at X + h; i.e. we want $k = \delta y$ such that $\left.\frac{dy}{dx}\right]_{X + h,\ Y + k} = f(X + h, Y + k)$.

We derive k by making four approximations and averaging them in a particular way.

First approximate the curve by a straight line through (X,Y) with the slope $\left.\frac{dy}{dx}\right]_{X,Y} = f(X,Y)$, and find a first approximation to k:

$$k_o = h \cdot f(X,Y)$$

Then we travel a fraction m of the way along this line to the point $(X + mh, Y + mk_o)$ and find $f(X + mh, Y + mk_o)$.

PX 71900-10-(178)

This gives us a new straight line through $(X + mh, Y + mk_o)$ with slope $f(X + mh, Y + mk_o)$, and we find

$$k_1 = h \cdot f(X + mh, Y + mk_o)$$

We now use k_o and k_1 to find a third point at which f is calculated: $(X + nh, Y + [n-r] k_o + rk_1)$.

$$k_2 = h\, f(X + nh, Y + [n-r] k_o + rk_1)$$

Similarly,

$$k_3 = h \cdot f(X + ph, Y + [p-s-t] k_o + sk_1 + tk_2$$

The weighted average of k_o, k_1, k_2, and k_3 is the desired $k = \delta y$:

$$\delta y = y(X + h) - y(X) = c_o k_o + c_1 k_1 + c_2 k_2 + c_3 k_3$$

where $c_o + c_1 + c_2 + c_3 = 1$.

For a system of equations, the same four steps given above are made for each equation and

$$\delta y_i = c_o k_{io} + c_1 k_{i1} + c_2 k_{i2} + c_3 k_{i3} \text{ where } c_o + c_1 + c_3 = 1.$$

The above process is, for certain values of m, n, p, s, t, c_o, c_1, c_2, and c_3, the Runge-Kutta process. The Gill process was derived, with application to machine use in mind, by minimizing the number of storage cells required. For the Gill Method the above constants are

$m = 1/2$, $r = 1 - \sqrt{1/2}$, $c_o = 1/6$

$n = 1/2$, $s = - \sqrt{1/2}$, $c_1 = (1/3)(1 - \sqrt{1/2})$

$p = 1$, $t = 1 + \sqrt{1/2}$, $c_2 = (1/3)(1 + \sqrt{1/2})$

$c_3 = 1/6$

The Gill process further systematizes the calculation so as to increase the accuracy and simplify the coding.

The Subroutine As used in the Gill Method Subroutine, the process is as follows:

1st pass:

Advance x by (1/2)h

$$k_{io} = h \cdot f_i(x, y_{10}, y_{20}, \ldots, y_{no})$$

$$r_{i1} = (1/2)k_{io} - q_{io}$$

$$q_{i1} = q_{io} + 3r_{i1} - (1/2)k_{io}$$

$y_{i1} = y_{io} + r_{i1}$

Calculate $f_i\ (x, y_{11}, y_{21}, \ldots, y_{n1})$ in programmer's own coding.

2nd pass:

$k_{i1} = h \cdot f_i\ (x, y_{11}, y_{21}, \ldots, y_{n1})$

$r_{i2} = (1 - \sqrt{1/2})\ (k_{i1} - q_{i1})$

$q_{i2} = q_{i1} + 3r_{i2} - (1 - \sqrt{1/2})\ k_{i1}$

$y_{i2} = y_{i1} + r_{i2}$

Calculate $f_i\ (x, y_{12}, y_{22}, \ldots, y_{n2})$ in programmer's own coding.

3rd pass:

Advance x by (1/2)h

$k_{i2} = h \cdot f_i\ (x, y_{12}, y_{22}, \ldots, y_{n2})$

$r_{i3} = (1 + \sqrt{1/2})\ (k_{i2} - q_{i2})$

$q_{i3} = q_{i2} + 3r_{i3} - (1 + \sqrt{1/2})\ k_{i2}$

$y_{i3} = y_{i2} + r_{i3}$

Calculate $f_i\ (x, y_{13}, y_{23}, \ldots, y_{n3})$ in programmer's own coding.

4th pass:

$k_{i3} = h \cdot f_i\ (x, y_{13}, y_{23}, \ldots\ y_{n3})$

$r_{i4} = (1/6)\ (k_{i3} - 2q_{i3})$

$q_{i4} = q_{i3} - 3r_{i4} - (1/2)k_{i3}$

$y_{i4} = y_{i3} + r_{i4}$

Calculate $f_i\ (x, y_{14}, y_{24}, \ldots, y_{n4})$ in programmer's own coding.

PX 71900-10-(178)

Errors The paper by S. Gill mentioned previously includes a detailed analysis of errors, both truncation error and round-off error.

The expression for the truncation error in δy_i is too complicated to give here, but its dominating term, the author states, is

$$\frac{h^5}{-120} \sum_{o}^{n} \left[f_j \frac{f_k}{y_j} \cdot \frac{f_m}{y_l} \cdot \frac{f_i}{y_m} \right]_{x=X} \quad \cdot \quad \text{where } y_o = x,\ f_o = 1, j, k, l, m$$

and the truncation error in δy_i will be approximately this when the second partial derivatives are all close to zero. It is probably more useful to say merely that the truncation error is of the order of h^5.

The standard deviation in $y_i - (1/3)\, q_i$ over one step from all rounding off errors is (where f is the quantity mentioned in the section on notation)

$$1/6 \left[7/3 \left\{ 2^{-2f} + (1/16)h^2 \sum_j \left(\frac{f_i}{y_j} \right)^2_X \right\} \right]^{1/2} u, \quad u = \text{the value of one unit in the last digit of } y.$$

Machine Checking

A driver routine was used to solve the following system of equations using the NUI-5 method:

$$\frac{dy_1}{dx} = y_2$$

$$\frac{dy_2}{dx} = -\alpha^2 y_1$$

This system is equivalent to the second order equation:

$$\frac{d^2y}{dx^2} + \alpha^2 y = 0$$

The initial conditions used were:

$$\text{AT}\quad X=0 \quad \begin{cases} y_1 = 1 \\ y_2 = -3i \end{cases}$$

α was taken as 3, and Δx as 0.0872664626 which is equivalent to five degrees.

The correct solution, $y_1 = \cos \alpha x - i \sin \alpha x$, when evaluated at $\alpha x = 90^\circ$ and at $\alpha x = 135^\circ$ yields $y_1 = 0 - 1i$ and $y_1 = 0.70710678 - 0.70710678\, i$.

The routine gave

$$y_1 = 0.000000044704 - 1i \text{ and}$$

$$y_1 = 0.70710672 - 0.70710684\, i \text{ at these values of } \alpha x.$$

Another driver routine using NUI-5 was utilized to solve the following systems of equations:

(1) $$\frac{dy_1}{dx} = y_1 a_{11} + y_2 a_{12} + y_3 a_{13} + y_4 a_{14}$$

(2) $$\frac{dy_2}{dx} = y_1 a_{21} + y_2 a_{22} + y_3 a_{23} + y_4 a_{24}$$

(3) $$\frac{dy_3}{dx} = y_1 a_{31} + y_2 a_{32} + y_3 a_{33} + y_4 a_{34}$$

(4) $$\frac{dy_4}{dx} = y_1 a_{41} + y_2 a_{42} + y_3 a_{43} + y_4 a_{44}$$

where the a's were the elements of the following matrix:

.38942862 -.056561159 i	.64293399	-.38130278	.31636930
.31354398 $\times 10^{-1}$	.51067378 -.23430743 i	.35343118	-.15651804
-.10142552 $\times 10^{-2}$	.63584848 $\times 10^{-1}$	.52424601 -.53041144 i	.28190968
-.91876338 $\times 10^{-3}$	-.57287441 $\times 10^{-2}$	.86142025 $\times 10^{-1}$	.52841129 -.94494028 i

PX 71900-10-(178)

At x = 0, the initial conditions were:

$$y_1 = 1 \qquad y_2 = 0 \qquad y_3 = 0 \qquad y_4 = 0$$

consequently, at x = 0 the four derivatives were equivalent to the first column of the matrix, respectively.

Two cases were calculated. Case 1, with $\Delta x = .1$ and case 2 with $\Delta x = .025$. The former case was checked by hand calculations.

The following table of the calculated results indicate that at x = .1, the results of all calculations agree to six decimal places.

A SUMMARY OF THE VALUES OF

y_1 , y_2 , y_3 AND y_4 FOR THE

HAND CALCULATION, CASE 1, AND CASE 2 AT x = .1

	Hand Calculation $\Delta x = .1$	Case 1 $\Delta x = .1$	Case 2 $\Delta x = .025$
x	.1	.1	.1
y_1	1.0398008 -.0058823261	1.0398000 -.0058818996i	1.0397999 -.0058819007i
y_2	.0032784372 -.0000477404901	.0032785024 -.0000477392191	.0032785024 -.0000477400411
y_3	-.000097036022 +.0000029048977i	-.000097038351 +.0000029047554i	-.000097038244 +.0000029045580i
y_4	-.000097412743 +.0000048894666i	-.000097414832 +.0000048844414i	-.000097414673 +.0000048843484i

D		GIL00	01024		02000	00	00000	00000
D		GIN00	01114		02132	00	00000	00000
D		GIM00	51071		65477	00	00000	00000
D		GCN00	51161		65631	00	00000	00000
GIM00	MS	00000	GIL00		65477	56	00000	02000
GIM01	MJ	00000	00000		65500	45	00000	00000
GIM02	MJ	00000	GIL59	SET UP ENTRA	65501	45	00000	02073
GIM03	MJ	00000	GIL10	1ST PASS	65502	45	00000	02012
GIM04	RA	GIN01	00016		65503	21	02133	00020
GIM05	EJ	GIN27	GIL79	EQLTY AGNST3	65504	43	02165	02117
GIM06	EJ	GIN28	GIL01	EQLTY AGNST5	65505	43	02166	02001
GIM07	TP	GIN32	00023	BRING IN DEL	65506	11	02172	00027
GIM08	TP	GIN33	00024	BRING IN DEL	65507	11	02173	00030
GIM09	MJ	00000	GIL24		65510	45	00000	02030
GIM10	TP	00016	GIN01	SET PASS CNT	65511	11	00020	02133
GIM11	TU	GIN02	GIL26	SET LOC A	65512	15	02134	02032
GIM12	RP	30004	GIL14	DELTA T IN23	65513	75	30004	02016
GIM13	TP	00000	00023	24 IN 25 26	65514	11	00000	00027
GIM14	TP	00023	GIN32	STORE DELTAT	65515	11	00027	02172
GIM15	TP	00024	GIN33	REAL / IMAG	65516	11	00030	02173
GIM16	LDAD	00023	00025	T PLUS DELTA	65517	14	30027	04031
GIM17	TP	00002	00037	STOR T PLUS	65520	11	00002	00045
GIM18	TP	00003	00038	DELTA T	65521	11	00003	00046
GIM19	TP	GIN03	00027	BRG ONE HALF	65522	11	02135	00033
GIM20	TP	GIN04	00028	TO 27 28	65523	11	02136	00034
GIM21	ADMP	00025	00027	GET T PLUS O	65524	14	04031	14033
GIM22	TP	00002	00000	STORE T PLUS	65525	11	00002	00000
GIM23	TP	00003	00000	ONE HALF DEL	65526	11	00003	00000
GIM24	TV	00000	GIN00	SET EQ COUNT	65527	16	00000	02132
GIM25	RP	30006	GIL27	A31 32 B33	65530	75	30006	02033
GIM26	TP	00000	00031	C 35 36	65531	11	00000	00037
GIM27	RA	GIL26	GIN29	SETABC FOR	65532	21	02032	02167
GIM28	TU	GIL78	GIL31	SET UP Y ADD	65533	15	02116	02037
GIM29	TV	GIL78	GIL54	REVERSE PREV	65534	16	02116	02066
GIM30	RP	30006	GIL32	Y25 26Y27 28	65535	75	30006	02040
GIM31	TP	00000	00025	Q 29 30	65536	11	00000	00031
GIM32	LDMP	00033	00029	BQ	65537	14	30041	14035
GIM33	TP	00002	00033	STORE BQ IN	65540	11	00002	00041
GIM34	TP	00003	00034	33 AND 34	65541	11	00003	00042
GIM35	LDMP	00025	00023	COMP Y DEL T	65542	14	30031	14027
GIM36	TP	00002	00023	STORE K IN	65543	11	00002	00027
GIM37	TP	00003	00024	23 AND 24	65544	11	00003	00030
GIM38	MPSU	00031	00033	AK BQ EQS R	65545	14	14037	10041
GIM39	TP	00002	00031	STORE R IN	65546	11	00002	00037
GIM40	TP	00003	00032	31 AND 32	65547	11	00003	00040
GIM41	ADNO	00027	00000	R PLUS 4	65550	14	04033	00000
GIM42	TP	00002	00027	NEW Y STORED	65551	11	00002	00033
GIM43	TP	00003	00028	IN 27 AND 28	65552	11	00003	00034
GIM44	LDMP	00023	00035		65553	14	30027	14043

```
GIM45    TP 00002 00035         STORE CK IN    65554  11 00002 00043
GIM46    TP 00003 00036         35 AND 36      65555  11 00003 00044
GIM47    TP GIN23 00033         BRING IN       65556  11 02161 00041
GIM48    TP 00013 00034         ONE THIRD      65557  11 00015 00042
GIM49  LDDV 00031 00033         COMPUTE 3R     65560  14 30037 20041
GIM50  ADSU 00029 00035                        65561  14 04035 10043
GIM51    TP 00002 00029         NEW Q STORED   65562  11 00002 00035
GIM52    TP 00003 00030         IN 29 AND 30   65563  11 00003 00036
GIM53    RP 30006 GIL55         YDAT IN25 26   65564  75 30006 02067
GIM54    TP 00025 00000         YIN27 28 QIN   65565  11 00031 00000
GIM55    RA GIL31 GIN29                        65566  21 02037 02167
GIM56    RA GIL54 GIN30                        65567  21 02066 02170
GIM57    RS GIN00 00016         SUB1 FMEQ CN   65570  23 02132 00020
GIM58    ZJ GIL83 00000                        65571  47 02123 00000
GIM59    TP GIL01 A0000         SET UP         65572  11 02001 20000
GIM60    LA A0000 00015                        65573  54 20000 00017
GIM61    TU A0000 GIL62         SET PARAMETE   65574  15 20000 02076
GIM62    TP 00000 A0000         PARA WORK A    65575  11 00000 20000
GIM63    TV A0000 GIL58         SET DER ENTR   65576  16 20000 02072
GIM64    TU A0000 GIL24         SET EQ C ADD   65577  15 20000 02030
GIM65    AT 00015 A0000         SET DEL T AD   65600  35 00017 20000
GIM66    TU A0000 GIL13                        65601  15 20000 02015
GIM67    LA A0000 00057                        65602  54 20000 00071
GIM68    AT GIN31 A0000                        65603  35 02171 20000
GIM69    TV A0000 GIL22         SET T REL AD   65604  16 20000 02026
GIM70    AT 00016 A0000                        65605  35 00020 20000
GIM71    TV A0000 GIL23         SET T IM ADD   65606  16 20000 02027
GIM72    AT 00016 A0000                        65607  35 00020 20000
GIM73    TV A0000 GIL78         SET LOC Y      65610  16 20000 02116
GIM74    LA A0000 00015                        65611  54 20000 00017
GIM75    TU A0000 GIL78         SET LOC Y      65612  15 20000 02116
GIM76    RA GIL01 00016         SET BY PAR W   65613  21 02001 00020
GIM77    MJ 00000 GIL01                        65614  45 00000 02001
GIM78    00 00000 00000 B       LOC OF Y       65615  00 00000 00000
GIM79  LDNO 00037 00000                        65616  14 30045 00000
GIM80    TP GIN32 00023         RESTORE DEL    65617  11 02172 00027
GIM81    TP GIN33 00024         IN 23 24       65620  11 02173 00030
GIM82    MJ 00000 GIL22                        65621  45 00000 02026
GIM83    TP GIN32 00023                        65622  11 02172 00027
GIM84    TP GIN33 00024                        65623  11 02173 00030
GIM85    TU GIL26 GIL88                        65624  15 02032 02130
GIM86    RS GIL88 GIN29                        65625  23 02130 02167
GIM87    RP 30006 GIL89                        65626  75 30006 02131
GIM88    TP 00000 00031                        65627  11 00000 00037
GIM89    MJ 00000 GIL30                        65630  45 00000 02036
GCN00    00 00000 00000 B       EQ COUNTER     65631  00 00000 00000
GCN01    00 00000 00000 B       PASS COUNTER   65632  00 00000 00000
GCN02    00 GIN03 00000 BRB      LOCATION A    65633  00 02135 00000
GCN03    05 00000 00000 -01 F      A1          65634  20 04000 00000
```

PX 71900-10-(173)

GCN04	00	00000	00000	B		A1	65635	00	00000	00000
GCN05	01	00000	00000	00	F	B1	65636	20	14000	00000
GCN06	00	00000	00000	B		B1	65637	00	00000	00000
GCN07	05	00000	00000	-01	F	C1	65640	20	04000	00000
GCN08	00	00000	00000	B		C1	65641	00	00000	00000
GCN09	02	92893	21881	-01	F	A2	65642	17	74537	30314
GCN10	00	00000	00000	B		A2	65643	00	00000	00000
GCN11	02	92893	21881	-01	F	B2	65644	17	74537	30314
GCN12	00	00000	00000	B		B2	65645	00	00000	00000
GCN13	02	92893	21881	-01	F	C2	65646	17	74537	30314
GCN14	00	00000	00000	B		C2	65647	00	00000	00000
GCN15	01	70710	67812	00	F	A3	65650	20	16650	11714
GCN16	00	00000	00000	B		A3	65651	00	00000	00000
GCN17	01	70710	67812	00	F	B3	65652	20	16650	11714
GCN18	00	00000	00000	B		B3	65653	00	00000	00000
GCN19	01	70710	67812	00	F	C3	65654	20	16650	11714
GCN20	00	00000	00000	B		C3	65655	00	00000	00000
GCN21	01	66666	66667	-01	F	A4	65656	17	65252	52525
GCN22	00	00000	00000	B		A4	65657	00	00000	00000
GCN23	03	33333	33333	-01	F	B4	65660	17	75252	52525
GCN24	00	00000	00000	B		B4	65661	00	00000	00000
GCN25	05	00000	00000	-01	F	C4	65662	20	04000	00000
GCN26	00	00000	00000	B		C4	65663	00	00000	00000
GCN27	00	00000	00003	B			65664	00	00000	00003
GCN28	00	00000	00005	B			65665	00	00000	00005
GCN29	00	00006	00000	B			65666	00	00006	00000
GCN30	00	00000	00006	B			65667	00	00000	00006
GCN31	00	00000	00002	B			65670	00	00000	00002
GCN32	00	00000	00000	B		FOR STORING	65671	00	00000	00000
GCN33	00	00000	00000	B		DELTA T	65672	00	00000	00000
START		00000						45	00000	00000

PX 71900-10-(178)

THE RAMO-WOOLDRIDGE CORPORATION
Los Angeles 45, California

Algebraic Equation Solver

Specifications

Identification Tag:	POL-O
Type:	Subroutine available on cards for assembly
Regions Used:	OOM, O1M O2M, OOK, FOO, COO. OOT
Storage:	157 words total program storage 2n + 17 cells temporary storage used immediately following the RW temporary pool where n is the order of the polynomial (See text) The constant pool and temporary storage pool are used by this routine
Entrance and Exit:	RJ SUBO1 SUBO2 results not punched RJ SUBO1 SUBO3 results punched (See text)
Alarm Exit:	A SNIP exponent overflow alarm may occur. (See text).
Machine Time:	See enclosed table.
Mode of Operation:	Floating Complex Arithmetic. SNIP must have been activated.

Coded by:	Werner L. Frank	May, 1956
Approved by:	W. F. Bauer	July, 1956

Description

This subroutine employs a new iterative procedure due to D. Muller[1] for finding the n roots of the polynomial equation

$$P(x) = \sum_{j=0}^{n} a_j x^{n-j} = 0$$

where the coefficients a_j can be real or complex and the roots need not be distinct. The routine operates using the complex arithmetic version of SNIP, finding the real and complex roots in an approximately ascending order of magnitude. As each root is obtained, the intermediate and final iterants can be punched out on cards, depending on the option chosen by the programmer. The polynomial is reduced one order after finding each root so that the k^{th} root is obtained from the polynomial $P_k(x)$ where

$$P_k(x) = \frac{P(x)}{\prod_{i=1}^{k-1}(x-x_i)} \qquad k > 1$$

In addition to the 157 words of storage needed by the subroutine, it is necessary to provide 2n + 17 cells of temporary storage immediately following the Ramo-Wooldridge Temporary Pool.

The polynomial coefficients must appear in succession beginning with the coefficient of the highest power of x, in a block of cells all of which must be in ES or all on the drum. Each coefficient occupies two cells, one for the real component and one for the complex. This code can theoretically solve polynomials of extremely large order. However, practical limitations are imposed by the available 1024 words of ES storage when using SNIP so that n must be less than 211.

PX 71900-10-(179)

---

[1] A paper "A Method for Solving Algebraic Equations using an Automatic Computer" by D. Muller is forthcoming in MTAC.

A serious deterioration in accuracy can result for polynomials with multiple roots and arbitrary polynomials of high degree. This is probably due to the low precision of the coefficients (27 binary bits) which necessarily define the roots. Also, for large degree, the range of numbers in SNIP ($\pm 2^{128}$) is often exceeded (see Alarm section). Hence, for problems of degree larger than 20, special care must be taken.

Operating Instructions

SNIP must have been activated, and be in ES.

The code provides for three options for output depending upon the entrance selected. In every case, the n roots of the polynomial P(x) can be found in the 2n cells starting at cell 50 = 62b when the subroutine is exited. The programmer is cautioned to note that the output hopper of SNIP is used regardless of the option selected. Assuming that the subroutine is assigned to region SUB00; that 00X00 is the location of the real part of the coefficient of the highest power of x in P(x) and that vvvvv is the degree, n, of the polynomial, then the options available to the programmer are:

Entrance No. 1a: RJ SUB01 SUB03
20 00X00 vvvvv

n sets of 3 cards each are punched each set containing the following information

Card No. 1	x_i	$P(x_i)$	$P'(x_i)$
Card No. 2	x_{i+1}	$\left\lvert \frac{P(x_i)}{x_{i+1} P'(x_i)} \right\rvert$	
Card No. 3	blank		

where x_{i+1} and x_i are the last two successive iterations for the root, real part in field one and imaginary part in field two.

PX 71900-10-179(

$$P(x_2) \sim a_n + a_{n-1} + a_{n-2}$$

$$P(x_3) = a_n$$

At each step of the iteration violent jumps in the function $P(x_{j+1})$ are prevented in order to provide a smooth convergence. This is accomplished by testing to see if

$$\left|\frac{P(x_{j+1})}{P(x_j)}\right| < 10$$

If this inequality does not hold then d_{j+1} is continually multiplied by 1/2 until an x^*_{j+1} is found sufficiently close to x_j so that the test is passed. In case the denominator of d_{i+3} is zero, unity is used in place of the denominator and the computation is resumed. The iteration continues until the convergence criterion

$$\left|\frac{x_{j+1} - x_j}{x_{j+1}}\right| < 10^{-m}$$

is met.

In this code m = 4. After an iterant x_{j+1} meets the above criterion one cycle of the second order Newton procedure is performed obtaining thereby the finally accepted approximation x_{j+2}

$$x_{j+2} = x_{j+1} - \frac{P(x_{j+1})}{P'(x_{j+1})}$$

This is approximately equivalent to a convergence criterion of 10^{-2m}.

After one root is obtained the polynomial is depressed one degree and another root is sought. Possible errors in successive roots, due to errors in the deflated polynomial, are eliminated by the final cycle of the Newton process which always refers back to the original polynomial. The code provides for the punching of successive iterants, their associated residues in the

$P(x_i)$ = residue of polynomial at x_i, components in field three and four respectively.

$P'(x_i)$ = derivative of polynomial at x_i, components in field five and six respectively

$\left|\frac{P(x_i)}{x_{i+1}P'(x_i)}\right|$ = measure of accuracy of root (see Mathematical Method) in field three.

Entrance No. 1b: RJ SUB01 SUB03

00 00X00 vvvvv

Each set of three cards is preceded by cards containing the values of all the successive iterants and their corresponding residues in the associated polynomial. These cards contain:

x_j $\qquad P_k(x_j)$

in fields one, two, three and four in addition to the results indicated for entrance No. 1a.

Entrance No. 2: RJ SUB01 SUB02

20 00X00 vvvvv

No cards are punched.

Alarm Conditions

It is possible to obtain a SNIP alarm for exponent overflow when the number range of $\pm 2^{128}$ is violated. It is advised in such instances that the problem be rescaled. This can be done by making the transformation $x = 10^k y$ or by dividing the coefficients of the polynomial by some factor. The latter procedure will yield a polynomial whose roots are equal to those of the original problem.

Mathematical Method

[illegible] procedure, due to D. Muller[1] of the University of Illinois, is employed in which quadratic fits are made to the polynomial

$$P(x) = \sum_{j=0}^{n} a_j x^{n-j} = 0$$

Given three arbitrary points, x_i, x_{i+1}, and x_{i+2}, we find the parabola passing through the points $P(x_i)$, $P(x_{i+1})$and $P(x_{i+2})$. Locating the root of this quadratic closest to x_{i+2}, say x_{i+3}, the process is repeated using now the points x_{i+1}, x_{i+2}, and x_{i+3} .

The explicit formulas used are:

$$x_{i+3} = x_{i+2} + (x_{i+2} - x_{i+1})\, d_{i+3}$$

where

$$d_{i+3} = \frac{-2P(x_{i+2})\,(1+d_{i+2})}{b_{i+2} \pm \left\{ b_{i+2}^2 - 4P(x_{i+2})d_{i+2}(1+d_{i+2})\left[P(x_i)d_{i+2} - P(x_{i+1})(1+d_{i+2}) + P(x_{i+2})\right]\right\}^{1/2}}$$

and

$$b_{i+2} = P(x_i)d_{i+2}^2 - P(x_{i+1})(1+d_{i+2})^2 + P(x_{i+2})(1+2d_{i+2})$$

The sign in the denominator is chosen in such a way so as to give d_{i+3} the smallest magnitude. Coupled with the starting procedure this finds the roots in an approximately ascending order of magnitude.

To start the process off, the following choice of values is made:

$$x_1 = -1$$

$$x_2 = 1$$

$$x_3 = 0$$

$$d_3 = -1/2$$

with the associated functional values approximating the polynomial $P(x)$ in the neighborhood of the origin:

$$P(x_1) \sim a_n - a_{n-1} + a_{n-2}$$

PX 71900-10-(179)

reduced polynomials and finally the five values x_j, $P(x_j)$, $P'(x_j)$, x_{j+1}, and M, where

$$M = \left| \frac{P(x_j)}{x_{j+1} P'(x_j)} \right|$$

This latter information gives an indication of the relative accuracy of the approximation x_{j+1}. Thus one should expect a value of $M = 10^{-7}$ or smaller for roots which are accurate to 6 places or more. Larger values of M indicate the possibility of less significance. In the event either x_{j+1} or $P'(x_j)$ are equal to zero no value for M is punched and $x_{j+1} = x_j$.

Local convergence of the Muller process has been shown to be somewhat less than second order for simple roots. In the case of multiple roots the order of convergence approaches a first order process. In this case the accuracy in the calculated roots deteriorates rapidly.

Machine Time

The time, in seconds, taken to find the n roots of a polynomial and punch the results according to the format described can be estimated from the following table of representative problems.

Degree n	Entrance No. 1a	Entrance No. 1b	Entrance No. 2
5	14	20	8
7	23	32	13
14	55	106	35
16	83	154	59
32	270	-	222
64	843	-	740

```
D               00M00  00098                      00142   00 00000 00000
D               01M00  00131                      00203   00 00000 00000
D               02M00  00161                      00241   00 00000 00000
D               00K00  00247                      00367   00 00000 00000
D               F0000  00002                      00002   00 00000 00000
D               C0000  00003                      00003   00 00000 00000
D               00T00  00033                      00041   00 00000 00000
00M00     MJ  00000  00000                        00142   45 00000 00000
00M01     MJ  00000  00000           EXIT         00143   45 00000 00000
00M02     MJ  00000  02M77           ENTRANCE 1   00144   45 00000 00356
00M03     TP  00K05  02M67           ENTRANCE 2   00145   11 00374 00344
00M04     SP  00M01  00015                        00146   31 00143 00017
00M05     TU  A0000  00M06                        00147   15 20000 00150
00M06     TP  00000  A0000                        00150   11 00000 20000
00M07     TU  A0000  00M18                        00151   15 20000 00164
00M08     TV  A0000  00K07                        00152   16 20000 00376
00M09     LA  A0000  00016                        00153   54 20000 00020
00M10     TP  00021  Q0000                        00154   11 00025 10000
00M11     QS  A0000  00M17                        00155   53 20000 00163
00M12     AT  02M17  A0000                        00156   35 00262 20000
00M13     QS  A0000  01M05                        00157   53 20000 00210
00M14     ST  00K06  A0000                        00160   36 00375 20000
00M15     TU  A0000  00M00                        00161   15 20000 00142
00M16     RA  00M17  00K03                        00162   21 00163 00372
00M17     RP  30000  00M19           TRANSFER     00163   75 30000 00165
00M18     TP  00000  00T15                        00164   11 00000 00060
00M19     RS  00K07  00016                        00165   23 00376 00020
00M20     TP  00K07  00T14                        00166   11 00376 00057
00M21     MJ  00000  00M28                        00167   45 00000 00176
00M22     TP  00013  00004                        00170   11 00015 00004
00M23     TP  00T14  00031                        00171   11 00057 00037
00M24   LDMP  00T15  00T06  B                     00172   14 32060 14047
00M25   ADNO  00T17  00000  BS                    00173   14 07062 00000
00M26     RA  00004  00K03                        00174   21 00004 00372
00M27     IJ  00031  00M24                        00175   41 00037 00172
00M28     RS  00M00  00K03                        00176   23 00142 00372
00M29     TU  00M00  00004                        00177   15 00142 00004
00M30     RS  01M05  00015                        00200   23 00210 00017
00M31     TU  01M05  01M06                        00201   15 00210 00211
00M32     RS  01M05  00015                        00202   23 00210 00017
01M00   LDAD  00000  00004  B    B                00203   14 32000 06004
01M01   SUST  00002  00T00  B        F0           00204   14 12002 34041
01M02   ADAD  00002  00002  B    B                00205   14 06002 06002
01M03     TP  F0000  00T02           F1           00206   11 00002 00043
01M04     TP  C0000  00T03                        00207   11 00003 00044
01M05     TP  00000  00T04                        00210   11 00000 00045
01M06     TP  00000  00T05           F2           00211   11 00000 00046
01M07     RP  10008  01M09                        00212   75 10010 00214
01M08     TP  00013  00T06            S           00213   11 00015 00047
01M09     TN  00K00  00T08             E          00214   13 00367 00051
```

PX 71900-10-(179)

01M10	TP	00K00	00T10	T	00215	11	00367	00053
01M11	TN	00K01	00T12		00216	13	00370	00055
01M12	PMNO	00000	00000	INITIAL	00217	14	24000	00000
01M13	ZJ	01M15	01M14		00220	47	00222	00221
01M14	TP	00K02	00T02	VALUES	00221	11	00371	00043
01M15	LDMP	00T00	00T08		00222	14	30041	14051
01M16	ADST	00T04	00023	E	00223	14	04045	34027
01M17	MPST	00T08	00025	V	00224	14	14051	34031
01M18	LDMP	00T02	00T10	A	00225	14	30043	14053
01M19	STSU	00027	00023	S L	00226	14	34033	11027
01M20	LDPM	00T04	00000	U	00227	14	30045	24000
01M21	ZJ	01M22	02M44	A	00230	47	00231	00315
01M22	LDMP	00T04	00T10	T	00231	14	30045	14053
01M23	ADNO	F0000	00000	E	00232	14	04002	00000
01M24	TN	F0000	00029		00233	13	00002	00035
01M25	TN	C0000	00030		00234	13	00003	00036
01M26	ADMP	F0000	00T08	ROOT	00235	14	04002	14051
01M27	MPLD	00023	00T04	S	00236	14	15027	30045
01M28	SUMP	00027	00T10		00237	14	10033	14053
01M29	ADMP	00025	F0000	S	00240	14	05031	14002
01M30	ADRT	00023	00023	S	00241	14	04027	51027
02M01	TN	C0000	C0000	CONJUGATE	00242	13	00003	00003
02M02	LDMP	F0000	00025		00243	14	30002	14031
02M03	SJ	02M04	02M06		00244	46	00245	00247
02M04	TN	00023	00023		00245	13	00027	00027
02M05	TN	00024	00024		00246	13	00030	00030
02M06	LDAD	00025	00023	S DENOMINATOR	00247	14	30031	05027
02M07	PMNO	00000	00000		00250	14	24000	00000
02M08	ZJ	02M10	02M09		00251	47	00253	00252
02M09	TP	00K01	00023		00252	11	00370	00027
02M10	LDDV	00029	00023		00253	14	30035	20027
02M11	TP	F0000	00T08		00254	11	00002	00051
02M12	TP	C0000	00T09		00255	11	00003	00052
02M13	MPST	00T12	00027		00256	14	14055	34033
02M14	ADST	00T06	00023		00257	14	04047	34027
02M15	TP	00K03	00004		00260	11	00372	00004
02M16	TP	00T14	00031		00261	11	00057	00037
02M17	MOLD	00T17	00T15		00262	14	00062	30060
02M18	MPAD	00023	00T15	B FORM	00263	14	14027	06060
02M19	RA	00004	00K03	FI 1	00264	21	00004	00372
02M20	IJ	00031	02M18		00265	41	00037	00263
02M21	TP	F0000	00025		00266	11	00002	00031
02M22	TP	C0000	00026		00267	11	00003	00032
02M23	DVPM	00T04	00000	RATE OF	00270	14	20045	24000
02M24	TJ	00K02	02M79	CHANGE IN	00271	42	00371	00360
02M25	TP	00K00	F0000	FI 1	00272	11	00367	00002
02M26	TP	00013	C0000		00273	11	00015	00003
02M27	LDMP	F0000	00T08	S	00274	14	30002	15051
02M28	MJ	00000	02M13		00275	45	00000	00256
02M29	ADNO	00T08	00000		00276	14	04051	00000
02M30	TP	F0000	00T10		00277	11	00002	00053
02M31	TP	00T09	00T11		00300	11	00052	00054

```
02M32    TP 00025 00T04                          00301  11 00031 00045
02M33    TP 00026 00T05                          00302  11 00032 00046
02M34  LDDV 00T12 00T06                          00303  14 30055 20047
02M35  PMNO 00000 00000                          00304  14 24000 00000
02M36    TJ 00K04 02M44        CONVERGED         00305  42 00373 00315
02M37    TP 02M34 A0000                          00306  11 00303 20000
02M38    TU 00M06 02M39                          00307  15 00150 00310
02M39    TJ 00000 01M15                          00310  42 00000 00222
02M40  LDST 00T04 00008                          00311  14 30045 34010
02M41  LDST 00T06 00006                          00312  14 30047 34006
02M42  PDNO 00010 00000         ITERANT          00313  14 74012 00000
02M43    MJ 00000 01M15                          00314  45 00000 00222
02M44    TP 00T06 00006        LOAD              00315  11 00047 00006
02M45    TP 00T07 00007          ROOT            00316  11 00050 00007
02M46    TU 00M18 00004        SET B BOX         00317  15 00164 00004
02M47    TP 00K07 00031        SET INDEX         00320  11 00376 00037
02M48    TP 00013 00025                          00321  11 00015 00031
02M49    TP 00013 00026                          00322  11 00015 00032
02M50  LDST 00000 00023  B     FORM              00323  14 32000 34027
02M51  LDMP 00025 00T06         RESIDUAL         00324  14 30031 14047
02M52  ADST 00023 00025          AND             00325  14 04027 34031
02M53  LDMP 00023 00T06         DERIVATIVE       00326  14 30027 14047
02M54  ADST F0000 00023  B                       00327  14 06002 34027
02M55    RA 00004 00K03                          00330  21 00004 00372
02M56    IJ 00031 02M51                          00331  41 00037 00324
02M57    RP 30004 02M59                          00332  75 30004 00334
02M58    TP 00023 00008                          00333  11 00027 00010
02M59  LDPM 00025 00000        N                 00334  14 30031 24000
02M60    ZJ 02M61 02M65         E                00335  47 00336 00342
02M61  LDNO 00023 00000          W               00336  14 30027 00000
02M62  DVSU 00025 00T06   S       T              00337  14 21031 10047
02M63    TN F0000 00T06            O             00340  13 00002 00047
02M64    TN C0000 00T07             N            00341  13 00003 00050
02M65    TU 00M00 00004         C                00342  15 00142 00004
02M66  LDST 00T06 00004     B    Y               00343  14 30047 36004
02M67  PDNO 00010 00000           C              00344  14 74012 00000
02M68  STPM 00006 00000            L             00345  14 34006 24000
02M69    ZJ 02M71 02M73             E            00346  47 00350 00352
02M70    MJ 00000 02M74                          00347  45 00000 00353
02M71  LDDV 00025 00T06                          00350  14 30031 20047
02M72  PMNO 00008 00000   S                      00351  14 25010 00000
02M73  PDPD 00010 00010                          00352  14 74012 74012
02M74    IJ 00T14 00M22                          00353  41 00057 00170
02M75    RA 00M01 00016                          00354  21 00143 00020
02M76    MJ 00000 00M01                          00355  45 00000 00143
02M77    TP 02M70 02M67                          00356  11 00347 00344
02M78    MJ 00000 00M04                          00357  45 00000 00146
02M79    TP 00027 00T12                          00360  11 00033 00055
02M80    TP 00028 00T13                          00361  11 00034 00056
02M81    TP 00023 00T06                          00362  11 00027 00047
```

```
02M82    TP 00024 00T07                      00363  11 00030 00050
02M83    TP 00K01 F0000                      00364  11 00370 00002
02M84    RP 30004 02M29                      00365  75 30004 00276
02M85    TP 00T02 00T00                      00366  11 00043 00041
00K00    05 00000 00000 -01 F   C            00367  20 04000 00000
00K01    01 00000 00000     F   O            00370  20 14000 00000
00K02    01 00000 00000   1 F     N          00371  20 45000 00000
00K03    00 00002 00000            S         00372  00 00002 00000
00K04    01 00000 00000 -04 F       T        00373  16 36433 34272
00K05  PDNO 00010 00000              A       00374  14 74012 00000
00K06    00 00004 00000               N      00375  00 00004 00000
00K07    00 00000 00000                TS    00376  00 00000 00000
START       00M00                            00000  45 00000 00142
```

L. P. [illegible]
C. [illegible]

CONVAIR
SAN DIEGO

UNPACKED FLOATING POINT CARD OUTPUT

DESCRIPTION:

This routine writes up to five decimal point numbers on cards from either ES or MD

Drum address 72013----72357 inc.

Number of instructions 270 octal

Constants 55 octal

Temporaries 76 octal

ES address (including temporaries) 01000--01442 inc.

Driver 71760--72012 inc.

The routine may be modified or may be used as a subroutine by the use of the driver, with ES stored starting at 74001.

All constants needed are included in the routine.

COMMAND SEQUENCE:

The subroutine is coded to start in cell 01000 and is entered by the write sequence

37	01001	01002
AB	UUUUU	VVVVV

In case of bull failure a start at PAK = 01000 will result in repeat of the interrupted instruction.

To use as a subroutine with the driver the sequence should be

37	71760	71761
AB	UUUUU	VVVVV

In case of bull failure a start at PAK = 72007 will restore ES and come to a 56 halt. Release of 56 stop will repeat interrupted instruction.

A. If A is equal to 4, 5, 6 or 7 the cards will not be numbered otherwise they will be numbered starting at one. Cell 01270 (72303) contains this counter and may be set at one less than the next positive card number desired.

B. Is not used.

UUUUU is the storage address of the first mantissa

VVVVV is the number of floating point numbers to be punched.

CARD FORM COLUMNS:

6, 21, 36, 51, 66 decimal point punched in card.

7-- 16, 22--31, 37--46, 52--61, 67--76 ten decimal digit mantissas

17, 32, 47, 62, 77 sign of the mantissas

18--19, 33--34, 48--49, 63--64, 78--79, exponents (power of ten) range -- 99 to 99

20, 35, 50, 65, 80 signs of exponent.

CODING NOTES:

No prime commands are included. 17 00000 72342 will advance a punch card.

In case of an unnormalized mantissa or an exponent with absolute value above 513 octal in any series to be punched, that number will be punched as negative zero. In addition an alarm listing of the first such address and mantissa will occur on the typewriter at the end of the last card and a 56 stop will occur which may be released to continue.

The maximum absolute value of exponents is 98 or 99 depending on the mantissa value.

The combined error of read in with IC011 and read out with IC012 will not exceed one in the tenth digit.

UNPACKED FLOATING POINT CARD OUTPUT

```
71760   00745   56 00000 30000
71761   00746   75 31777 71763      STORE
71762   00747   11 00001 74001        E S
71763   00750   75 30400 00752      ROUTINE
71764   00751   11 71760 00745        TO E S
71765   00752   31 00745 00000      SET
71766   00753   36 01310 72011       REPEAT
71767   00754   16 00745 00772      SET
71770   00755   21 00772 01310       EXIT
71771   00756   55 00745 20025      MODIFY
71772   00757   44 00761 00760       E S
71773   00760   32 00777 00000         ADDRESSES
71774   00761   55 20000 00017          TO STORAGE
71775   00762   37 00762 00763
71776   00763   16 20000 00764      ACQUIRE
71777   00764   71 01310 30000       CONTROL WORD
72000   00765   55 20000 00006      EXAMINE
72001   00766   37 00762 00757       FOR E S
72002   00767   55 20000 00017         ADDRESSES
72003   00770   37 01001 01006      TO SUBROUTINE
72004   00771   11 01270 72303      STORE CARD NUMBER
72005   00772   75 31777 30000      RESTORE
72006   00773   11 74001 00001         E S
72007   00774   75 31777 72011      RESTORE
72010   00775   11 74001 00001       E S
72011   00776   30 00000 00000         REPEAT
72012   00777   74 00000 00000      E S STORAGE
72013   01000   30 00000 0000*      REPEAT LAST INST
72014   01001   45 00000 30000      EXIT
72015   01002   16 01001 01005
72016   01003   21 01001 01310
72017   01004   36 01344 01000
```

FX 71900-10-(180)

UNPACKED FLOATING POINT CARD OUTPUT

```
72020  01005  71 01310 30000
72021  01006  15 20000 01026    FIRST ADDRESS
72022  01007  13 20000 01347    FLAG
72023  01010  11 01310 01350    FLAG
72024  01011  31 20000 00025    EXTRACT
72025  01012  31 20000 00063     AND STORE
72026  01013  36 01310 01351      N-1
72027  01014  46 01001 01015    TEST FOR ZERO
72030  01015  17 00000 01327    PICK WRITE CARD
72031  01016  31 01351 00000
72032  01017  11 20000 01352    N-1 PER CARD
72033  01020  11 01325 10000    WRITE
72034  01021  42 01342 01024
72035  01022  11 01306 01352    5 PER CARD
72036  01023  11 01324 10000    WRITE AND PICK CARD
72037  01024  17 00000 10000    CARD INSTRUCTION
72040  01025  75 30012 01027    5 WORDS TO
72041  01026  11 30000 01353     STORAGE
72042  01027  75 10045 01031    CLEAR FOR
72043  01030  11 01307 01376     CARD IMAGE
72044  01031  15 01266 01050    SET ACQUISITION
72045  01032  11 01340 01374    BIT
72046  01033  11 01265 01375    BIT INSTRUCTION
72047  01034  41 01347 01047    IDENT NUMBER TEST
72050  01035  11 01332 01374    BIT
72051  01036  21 01270 01310    COUNT CARDS
72052  01037  42 01315 01041    SIZE OF WORD
72053  01040  31 01341 00000    SET 99999
72054  01041  32 01307 00043    INTEGER
72055  01042  32 01265 00000     TO
72056  01043  73 01315 01365      FRACTION
72057  01044  54 01365 00001       2 EXP 36
```

UNPACKED FLOATING POINT CARD OUTPUT

72060	01045	11	01306	01376	N-1 DIGITS
72061	01046	37	01200	01165	IDENT NUMBER
72062	01047	75	30002	01051	MANTISSA
72063	01050	11	30000	01365	AND EXPONENT
72064	01051	11	01365	01370	MANTISSA SIGN FLAG
72065	01052	54	01366	00030	EXPONENT
72066	01053	54	01366	00060	EXTENSION
72067	01054	12	20000	20000	EXPONENT
72070	01055	42	01301	01067	FLOATING TEST
72071	01056	41	01350	01062	FIRST TIME
72072	01057	11	01264	01370	MANTISSA FLAG
72073	01060	75	10002	01156	SET EQUAL
72074	01061	11	01307	01365	TO ZERO
72075	01062	13	01322	01350	SET ALARM FLAG
72076	01063	11	01026	01345	
72077	01064	15	01050	01065	ACQUIRE
72100	01065	11	30000	01346	MANTISSA
72101	01066	45	00000	01057	
72102	01067	11	01365	20000	
72103	01070	47	01071	01156	ZERO MANTISSA
72104	01071	46	01072	01073	
72105	01072	13	01365	01365	NEGATIVE MANTISSA
72106	01073	16	01323	01123	SET SHIFT TO 33
72107	01074	32	01307	00001	EXP 36
72110	01075	43	20000	01056	FLOATING TEST
72111	01076	11	01307	01373	CLEAR FOR DECIMAL EXP
72112	01077	11	01366	20000	EXPONENT
72113	01100	46	01101	01117	SIGN
72114	01101	23	01373	01342	10 EXP 5 ADJUSTMENT
72115	01102	11	01307	01372	CLEAR FOR 74
72116	01103	71	01315	01365	X 10 EXP 5
72117	01104	74	20000	01372	

UNPACKED FLOATING POINT CARD OUTPUT

72120	01105	11	20000	01365	
72121	01106	46	01107	01114	TEST FOR ROUND
72122	01107	21	01365	01310	ROUND
72123	01110	43	20000	01114	MANTISSA
72124	01111	31	01365	00107	AND
72125	01112	11	20000	01365	ADJUST
72126	01113	21	01372	01310	
72127	01114	21	01366	01372	ADJUST BSF
72130	01115	46	01101	01116	SIGN OF BSF
72131	01116	31	01366	00000	DIVIDE EXP
72132	01117	73	01323	01367	BY 33
72133	01120	11	20000	01366	EXPONENT REMAINDER
72134	01121	55	01365	00001	MANTISSA
72135	01122	45	00000	01152	
72136	01123	31	01365	30000	MANTISSA SHIFT
72137	01124	11	20000	01371	STORE AR
72140	01125	34	20000	00044	ERASE
72141	01126	11	20000	01365	STORE AL
72142	01127	75	20013	01426	TEST FOR LARGER
72143	01130	42	01310	01131	POWER OF TEN
72144	01131	51	01330	10000	
72145	01132	31	01311	00000	10 EXP 10
72146	01133	36	10000	10000	
72147	01134	35	01373	01373	INCREASE EXP
72150	01135	31	10000	00017	N X 2 EXP 15
72151	01136	35	01263	01137	
72152	01137	30	00000	[illegible]	ACQUIRE DIVISOR
72153	01140	11	[illegible]	[illegible]	
72154	[illegible]	[illegible]	[illegible]	[illegible]	RESTORE
					A
					2 EXP 35
					2 EXP 36

UNPACKED FLOATING POINT CARD OUTPUT

72160	01145	31	20000	00001	DETERMINE
72161	01146	42	01372	01150	LAST
72162	01147	27	10000	01310	BIT
72163	01150	11	10000	01365	ANSWER X 2 EXP 36
72164	01151	37	01151	01152	
72165	01152	41	01367	01123	HIGHER ORDER DIGIT
72166	01153	16	01366	01123	LOWER ORDER SHIFT
72167	01154	37	01151	01123	LOWER ORDER DIGIT
72170	01155	11	01373	01366	DECIMAL EXPONENT
72171	01156	21	01050	01326	STEP
72172	01157	21	01026	01326	STEP
72173	01160	37	01200	01174	SHIFT FOR PERIOD
72174	01161	12	01366	20000	EXPONENT
72175	01162	11	01333	01376	TALLY
72176	01163	73	01311	01372	DIGITS
72177	01164	11	20000	01371	OF EXP
72200	01165	31	01365	00002	EXTRACT
72201	01166	32	01365	00001	AND
72202	01167	11	20000	01365	POSITION
72203	01170	34	20000	00063	DIGIT
72204	01171	35	01375	01172	ASSEMBLE INST
72205	01172	30	00000	0000*	SET BIT
72206	01173	37	01173	01174	
72207	01174	55	01374	00043	SHIFT BIT
72210	01175	44	01176	01177	
72211	01176	21	01375	01334	ADVANCE FIELD
72212	01177	41	01376	01165	TEST FOR END
72213	01200	37	01200	01201	
72214	01201	11	01370	10000	
72215	01202	44	01203	01205	
72216	01203	13	01331	20000	
72217	01204	37	01173	01171	SIGN

UNPACKED FLOATING POINT CARD OUTPUT

72220	01205	55	01374	00043	SHIFT BIT
72221	01206	37	01206	01207	
72222	01207	31	01372	00017	TENS DIGIT
72223	01210	37	01200	01171	
72224	01211	31	01371	00017	UNITS DIGIT
72225	01212	37	01200	01171	
72226	01213	11	01366	01370	
72227	01214	37	01206	01201	
72230	01215	37	01215	01216	
72231	01216	41	01352	01047	WORDS PER CARD
72232	01217	11	01335	01371	SIT FOR 12 ROWS
72233	01220	21	01377	01336	
72234	01221	21	01411	01336	
72235	01222	21	01404	01336	
72236	01223	21	01413	01337	
72237	01224	21	01425	01337	
72240	01225	21	01420	01337	
72241	01226	16	01267	01233	
72242	01227	16	01127	01234	
72243	01230	15	01267	01231	
72244	01231	55	30000	00010	
72245	01232	77	00000	10000	
72246	01233	77	10000	30000	
72247	01234	77	10000	30000	
72250	01235	23	01231	01331	
72251	01236	23	01233	01310	
72252	01237	23	01234	01310	
72253	01240	41	01371	01231	TEST FOR END OF CARD
72254	01241	23	01351	01342	TEST FOR
72255	01242	46	01243	01016	END
72256	01243	41	01350	01001	FLOATING FLAG
72257	01244	75	10005	01246	

UNPACKED FLOATING POINT CARD OUTPUT

72260	01245	61	00000	01302	
72261	01246	11	01306	01371	
72262	01247	55	01345	10011	
72263	01250	31	01264	00000	
72264	01251	52	01343	01252	
72265	01252	30	00000	00000	
72266	01253	55	10000	00003	
72267	01254	41	01371	01250	
72270	01255	37	01255	01256	
72271	01256	55	01346	10003	
72272	01257	61	00000	01306	
72273	01260	11	01335	01371	
72274	01261	37	01255	01250	
72275	01262	56	00000	01001	
72276	01263	55	01310	10001	
72277	01264	61	00000	01271	
72300	01265	21	01401	01374	
72301	01266	00	01353	00000	
72302	01267	00	01442	01412	
72303	01270	00	00000		CARD COUNTER
72304	01271	00	00000	00037	
72305	01272	00	00000	00052	
72306	01273	00	00000	00074	
72307	01274	00	00000	00*70	
72310	01275	00	00000	00064	
72311	01276	00	00000	00062	
72312	01277	00	00000	00066	
72313	01300	00	00000	72	
72314	01301	00	00000	00513	
72315	01302	00	00000	00*45	
72316	01303	00	00000	00031	
72317	01304	00	00000	00020	

PX 71900-10-(180)

UNPACKED FLOATING POINT CARD OUTPUT

72320	01305	00	00000	0***1
72321	01306	00	00000	00*04
72322	01307	00	00000	00 *
72323	01310	00	00000	1
72324	01311	00	00000	00012
72325	01312	00	00000	00144
72326	01313	00	00000	01750
72327	01314	00	00000	23420
72330	01315	00	00003	03240
72331	01316	00	00036	41100
72332	01317	00	00461	13200
72333	01320	00	05753	60400
72334	01321	00	73465	45000
72335	01322	11	24027	62000
72336	01323	00	00000	00041
72337	01324	00	00000	00112
72340	01325	00	00000	00102
72341	01326	00	00002	00000
72342	01327	00	00000	00110
72343	01330	00	00000	00 77
72344	01331	00	00001	00000
72345	01332	40	00000	00000
72346	01333	00	00000	00011
72347	01334	00	00014	00000
72350	01335	00	00000	00013
72351	01336	01	00001	00001
72352	01337	00	00100	00100
72353	01340	01	00000	00000
72354	01341	00	00003	03237
72355	01342	00	00000	00*05
72356	01343	00	00000	00 *7
72357	01344	00	00000	00 *2

CONTINUOUS MATRIX MULTIPLIER USING FLIP III

This routine multiplies two matrices, takes this product times a third matrix, the resulting product times a fourth, etc. All input and output data is by means of six-field flip cards. The elements of any one of the matrices may be punched on cards with the elements of a row or a column in consecutive order with each row or column starting at the beginning of a flip card.* The multiplier assumes the matrix on the left is stored by rows and the one on the right is stored by columns. Even though one or both of the matrices may not be of this form on the cards, the multiplication may still be performed by using flags to indicate storage form different from the card form. A third flag is used to indicate a change in the form of the product if it is desired, and a fourth flag is used to indicate further multiplication. These flags are denoted by the symbols p_1, p_2, p_3, and p_4. Three parameters, m, n, and s, are necessary to furnish the routine with the size of the matrices.

To illustrate the use of the parameters and flags, suppose the product, ABC, of three matrices is wanted where the matrices have the following properties:

A : a $m \times n$ matrix, punched by columns on flip cards

B : a $n \times s$ matrix, punched by columns on flip cards

C : a $s \times t$ matrix, punched by rows on flip cards.

* The stop read indicator (a 12 - punch in column 1) <u>must</u> be on the last card for each matrix.

The multiplication proceeds from the left, so AB is the first product formed, then this multiplies C.

Before reading matrix A, the routine reads three parameter cards. The first of these parameter cards contains "m" as the card number, the second contains "n" as the card number, and the third contains "s" as the card number. The third card also contains any of the four flags that need to be indicated. If flag p_1 is necessary, it appears as any number different from zero in field 1 and so on for the other three flags in the next three fields. Thus, for this multiplication $p_1 \neq 0$ since A needs to be changed in form before storage, $p_2 = 0$ since B does not need to be changed, $p_3 = 0$ since AB is multiplying C instead of the transpose of AB times C, and $p_4 \neq 0$ since the multiplication will not terminate with the product AB. After the product AB has been computed by columns, it is stored by rows in the position of the first matrix in the multiplier, and the routine is ready to receive C. However, a new parameter card must be read before reading C. This card contains the second dimension of C, i. e. "t", as the card number, nothing in field 1 since the first matrix in the multiplier has been properly stored, $p_2 \neq 0$ in field 2 since the form of C must be changed from cards to storage, $p_3 = 0$ since the product of ABC is wanted not its transpose, and $p_4 = 0$ since the multiplication is to terminate when ABC is formed.

To summarize,

p_1 ($= 0$ for first matrix punched by rows
($\neq 0$ for first matrix punched by columns

p_2 ($= 0$ for second matrix punched by columns
($\neq 0$ for second matrix punched by rows

p_3 ($= 0$ for the product to be used as formed
($\neq 0$ for the transpose of the product to be used

p_4 (= 0 for no further multiplication
(≠ 0 for continued multiplication

NOTE: If $p_4 = 0$, then p_3 controls the form of the output.

The final output is the product by columns for $p_3 = 0$, or is the transpose (of the product) by columns for $p_3 \neq 0$. In the routine, just prior to the final output, the only MS - 1 appears in the program. If a stop is made here and MJ - 1 is set, then the output is that specified by p_3 and also its transpose.

In a continuous multiplication, output of the intermediate products may be obtained. The only MS - 2 in the routine is just prior to the intermediate output. A stop at this point allows the operator to set the MJ 's which control the intermediate output as follows:

MJ 's	Output
none	none
1	product by columns for $p_3 = 0$, transpose (of product) by columns for $p_3 \neq 0$.
2	Product regardless of p_3
3	Transpose (of Product) regardless of p_3
1 & 2	transpose (of product) for $p_3 = 0$ Product for $p_3 \neq 0$.
1 & 3	Same as 1.
2 & 3	Product followed by transpose for $p_3 = 0$ Transpose followed by product for $p_3 \neq 0$.
1, 2, & 3	Same as 1 & 2.

This routine, the working storage, the matrix storage and flip use

PX 71900-10-(181)

the entire ES storage and all of MD. The storage is allotted as follows:

00000 - 00077	Flip
01477 - 01777	Flip
70000 - 77777	Flip
00100 - 00137 -	Temporaries
00140 - 00157 -	Constants
00160 - 00376 -	Program
00377 - 01476 -	Working storage
40000 - 40627 -	Program
40630 - 50417 -	1st matrix
50420 - 60207 -	2nd matrix
60210 - 67777 -	Product

Tests are built into the routine that give an alarm exit if the size of a matrix is too large for the multiplier. The following restrictions are imposed on m, n, s, k_1, k_2, & k_3 (see next page for definition of k_i 's):

$$1 \leq m,\ n,\ s \leq 192_{10}$$
$$1 \leq m \times k_2 \leq 3960 = 60(66) + 1$$
$$1 \leq k_2 \times s \leq 3960 = 60(66) + 1$$
$$1 \leq m \times k_3 \leq 3960 = 60(66) + 1$$
$$1 \leq k_1 \times s \leq 3960 = 60(66) + 1$$

Thus, the largest square matrix that can be handled is a 60×60 matrix, and the largest rectangular matrix is a 60×66. The largest square matrix is a 60×60 since a 61×61 would be handled on cards and in the memory as a 61×66 or 66×61 and the product $61 \times 66 = 4026$ which is greater than the 3960 number of elements allowed.for each matrix.

A check sum is printed at the beginning of the program. This check sum is equal to 224205235612. A programmed Blotto is included in the routine. Setting (PAK) = 40620 and starting gives a Blotto of 00100 - 00377 and 40000 to 40637, which includes the temporaries, constants, and program in E.S. and entire program on M.D.

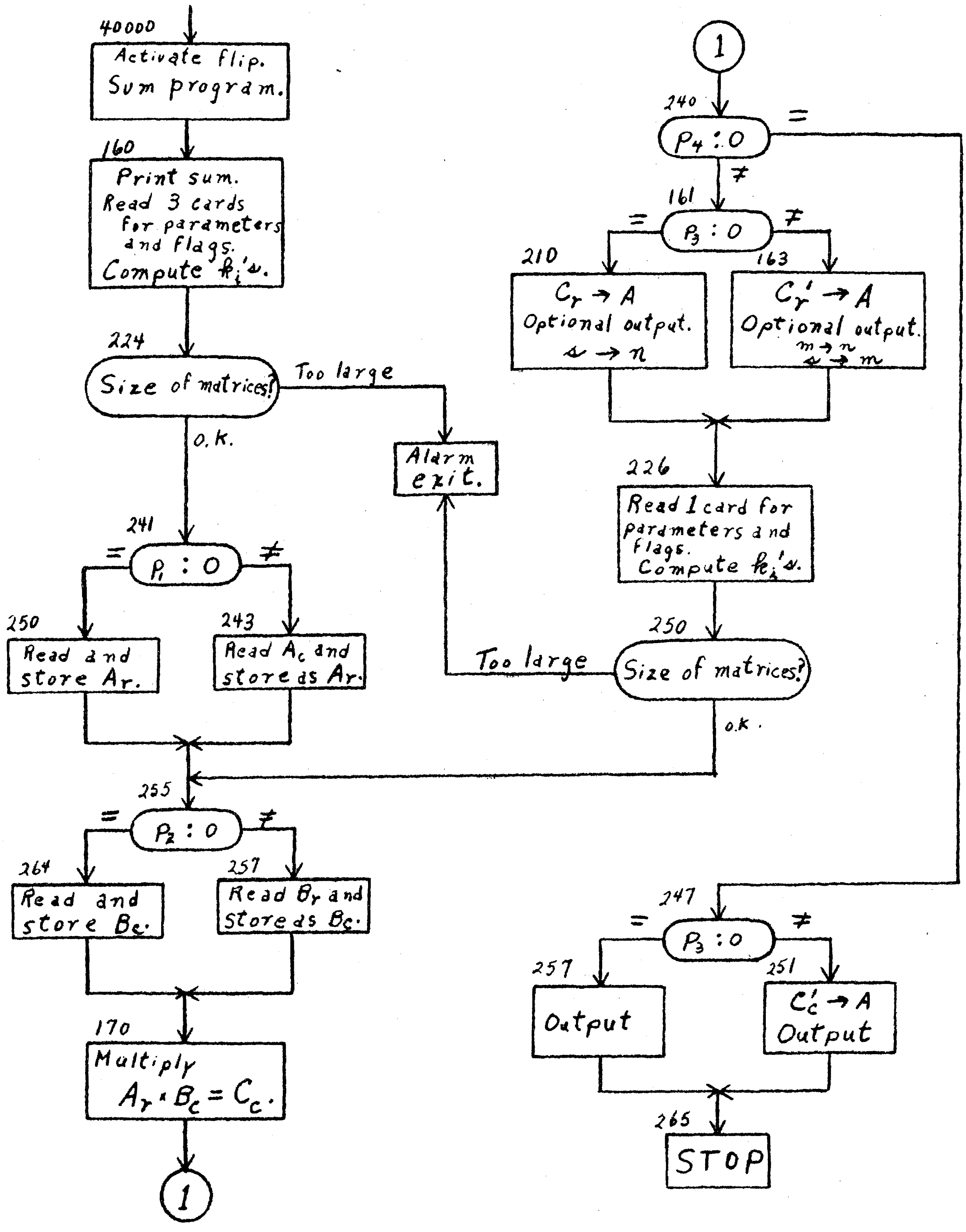
The program starts with (PAK) = 40000. Several continuous multiplication may be done by resetting (PAK) = 40000 at MS-0. However, either the cards should be loaded in the reading hopper when PAK is reset; or, all cards may be loaded beforehand if a blank card is placed between the cards for the separate continuous multiplications.

List of Symbols in Flow Chart and Code.

A	Matrix on left in product
B	Matrix on right in product
C	Product of A and B
a_{ij}	Element of A in ith row, jth column (refers to M.D. storage)
b_{jk}	Element of B in jth row, kth column (refers to M.D. storage)
c_{ik}	Element of C in ith row, kth column (refers to M.D. storage)
m	Number of rows of A, number of rows of C
n	Number of columns of A and number of rows of B
s	Number of columns of B, number of columns of C
k_1	m, if $m \equiv 0 \pmod 6$; 1st multiple of 6 $> m$ if $m \not\equiv 0 \pmod 6$
k_2	n, if $n \equiv 0 \pmod 6$; 1st multiple of 6 $> n$ if $n \not\equiv 0 \pmod 6$
k_3	s, if $s \equiv 0 \pmod 6$; 1st multiple of 6 $> s$ if $s \not\equiv 0 \pmod 6$
p_1	non-zero for A changed from cards to storage, otherwise zero
p_2	non-zero for B changed from cards to storage, otherwise zero
p_3	non-zero for transpose of product wanted, otherwise zero

p_4	non-zero for additional multiplication to do, otherwise zero
I_1	contents of a temporary storage cell in E. S.
d_{ij}	any matrix element on MD
δ_{ij}	any matrix element in ES after card read when p_1 or $p_2 \neq 0$
γ_j	one of a consecutive group of elements which is formed in ES before transfer to M.D. The group is a row of A or a column of B on card read with p_1 or $p_2 \neq 0$, is a column of C during multiplication, and is a row of C if $p_4 \neq 0$, $p_3 = 0$, or if $p_4 = 0$, $p_3 \neq 0$.
α_j	an element of the ith row of A
β_j	an element of the kth column of B
l ( )	location of quantity in parenthesis
A_r, B_r, C_r	indicated matrix stored by rows
A_c, B_c, C_c	indicated matrix stored by columns

GENERAL FLOW CHART



PX 71900-10-(181)

CONTINUOUS MATRIX MULTIPLIER

40000		45 00000 40001	40001 → (PAK)
40001		37 70160 70140	ACTIVATE FLIP
40002		00 00000 00000	FLAG
40003		31 40000 00000	CHECK
40004		75 20627 40006	SUM
40005		32 40001 00000	PROGRAM
40006		11 20000 10000	SUM → (Q)
40007		75 30237 00160	STARTER AND CARD READ
40010		11 40020 00140	TO ES, 160 → (PAK)
40011		00 00000 00000	
40012		00 00000 00000	
40013		00 00000 00000	
40014		00 00000 00000	
40015		00 00000 00000	
40016		00 00000 00000	
40017		00 00000 00000	
40020	00140	00 40630 40630	$\ell(a_{11}) \times 2^{15} + \ell(a_{11})$
40021	00141	00 50420 50420	$\ell(b_{11}) \times 2^{15} + \ell(b_{11})$
40022	00142	00 60210 60210	$\ell(c_{11}) \times 2^{15} + \ell(c_{11})$
40023	00143	00 00377 01177	$\ell(\delta_{11}) \times 2^{15} + \ell(\delta_{11})$
40024	00144	00 00106 00100	$\ell(m) \times 2^{15} + \ell(k_1)$
40025	00145	00 00000 00103	$\ell(k_1 \times 2^{15})$
40026	00146	00 07777 00000	4 DIGIT U-EXTRACTOR
40027	00147	00 00077 00000	2 DIGIT U-EXTRACTOR
40030	00150	00 00000 00006	6
40031	00151	00 00000 00572	378
40032	00152	00 00000 07571	3961
40033	00153	00 00000 00301	193
40034	00154	00 00000 00000	
40035	00155	00 00000 00000	
40036	00156	00 00000 00000	
40037	00157	00 00000 00000	

CONTINUOUS MATRIX MULTIPLIER

40040	00160	11	00057	00121	PRINT
40041	00161	61	00000	00042	
40042	00162	55	10000	00003	OF
40043	00163	51	00067	20000	
40044	00164	35	00042	00165	CHECK
40045	00165	30	00000	00000	
40046	00166	41	00121	00162	SUM
40047	00167	17	00000	73376	READY CARD REPRODUCER
40050	00170	1457	1010	0106	READ $m \rightarrow (106)$
40051	00171	1457	1010	0107	READ $n \rightarrow (107)$
40052	00172	1457	1010	0110	READ $s \rightarrow (110)$ FOLLOWED BY FLAGS
40053	00173	11	00106	20000	$m \rightarrow (A)$
40054	00174	42	00153	00176	IS $193 > m$?
40055	00175	37	76000	76001	NO. ALARM EXIT
40056	00176	11	00107	20000	YES. $n \rightarrow (A)$
40057	00177	42	00153	00201	IS $193 > n$?
40060	00200	37	76000	76001	NO. ALARM EXIT
40061	00201	11	00110	20000	YES. $s \rightarrow (A)$
40062	00202	42	00153	00204	IS $193 > s$?
40063	00203	37	76000	76001	NO. ALARM EXIT
40064	00204	11	00041	00121	YES. SET INDEX
40065	00205	15	00144	00210	$\ell(m) \rightarrow \ell(m, n, s)$
40066	00206	16	00144	00215	$\ell(k_1) \rightarrow \ell(k_i)$
40067	00207	16	00145	00217	$\ell(k_1 \times 2^{15}) \rightarrow \ell(k_i \times 2^{15})$
40070	00210	11	30000	20000	$m, n,$ OR $s \rightarrow (A)$
40071	00211	73	00150	10000	$(A)/6 \rightarrow (Q)$, REM $\rightarrow (A)$
40072	00212	47	00213	00214	IS $(A) = 0$?
40073	00213	21	10000	00074	NO. $(Q) + 1 \rightarrow (Q)$
40074	00214	71	10000	00150	YES. $(Q) \times 6 \rightarrow (A)$
40075	00215	11	20000	30000	$(R) \rightarrow k_i$
40076	00216	54	20000	00017	$(A) \times 2^{15} \rightarrow (A)$
40077	00217	11	20000	30000	$(R) \rightarrow k_i \times 2^{15}$

PX 71900-10-(181)

CONTINUOUS MATRIX MULTIPLIER

40100	00220	21	00210	00073	$\ell(m,n,s)+1 \rightarrow \ell(m,n,s)$
40101	00221	21	00215	00074	$\ell(k_i)+1 \rightarrow \ell(k_i)$
40102	00222	21	00217	00074	$\ell(k_i \times 2^{15})+1 \rightarrow \ell(k_i \times 2^{15})$
40103	00223	41	00121	00210⌋	TEST INDEX
40104	00224	71	00106	00101	$m \times k_2 \rightarrow (A)$
40105	00225	42	00152	00227	IS 3961 $> m \times k_2$?
40106	00226	37	76000	76001	NO. ALARM EXIT
40107	00227	71	00101	00110	YES. $k_2 \times s \rightarrow (A)$
40110	00230	42	00152	00232	IS 3961 $> k_2 \times s$?
40111	00231	37	76000	76001	NO. ALARM EXIT
40112	00232	71	00100	00110	YES. $k_1 \times s \rightarrow (A)$
40113	00233	42	00152	00235	IS 3961 $> k_1 \times s$?
40114	00234	37	76000	76001	NO. ALARM EXIT
40115	00235	71	00106	00102	YES. $m \times k_3 \rightarrow (A)$
40116	00236	42	00152	00240	IS 3961 $> m \times k_3$?
40117	00237	37	76000	76001	NO. ALARM EXIT
40120	00240	56	30000	00241	YES. MS→3
40121	00241	11	00111	20000	$p_1 \rightarrow (A)$
40122	00242	47	00243	00250	IS $p_1 = 0$? YES. 250 → (PAK)
40123	00243	11	00100	00121	NO. $k_1 \rightarrow I_1$
40124	00244	11	00106	00122	$m \rightarrow I_2$
40125	00245	11	00140	00123	$\ell(a_{11}) \rightarrow I_3$
40126	00246	37	00300	00301	CHANGE A_C TO A_λ AND STORE
40127	00247	45	00000	00255	255 → (PAK)
40130	00250	16	00140	00251	$\ell(a_{11}) \rightarrow \ell(a_{ij})$
40131	00251	⌈1457	0773	0000	READ 63 CARDS INTO $\ell(a_{ij})$ ETC.
40132	00252	21	00251	00151	$\ell(a_{ij}) + 378 \rightarrow \ell(a_{ij})$
40133	00253	11	00004	20000	(00004) → (A)
40134	00254	47	00255	00251⌋	IS (A) = 0 ? YES. 251 → (PAK)
40135	00255	11	00112	20000	NO. $p_2 \rightarrow (A)$
40136	00256	47	00257	00264	IS $p_2 = 0$? YES. 264 → (PAK)
40137	00257	11	00102	00121	NO. $k_3 \rightarrow I_1$

CONTINUOUS MATRIX MULTIPLIER

40140	00260	11	00110	00122	$A \to I_2$
40141	00261	11	00141	00123	$\ell(b_{11}) \to I_3$
40142	00262	37	00300	00301	CHANGE B_r TO B_c AND STORE
40143	00263	45	00000	00274	274 → (PAK)
40144	00264	16	00141	00265	$\ell(b_{11}) \to \ell(b_{ij})$
40145	00265	1457	0773	0000	READ 63 CARDS INTO $\ell(b_{ij})$ ETC.
40146	00266	21	00265	00151	$\ell(b_{ij}) + 378 \to \ell(b_{ij})$
40147	00267	11	00004	20000	(00004) → (A)
40150	00270	47	00274	00265	IS (A) = 0? YES. 265 → (PAK)
40151	00271	00	00000	00000	
40152	00272	00	00000	00000	
40153	00273	00	00000	00000	
40154	00274	75	30217	00170	NO. MULTIPLIER AND
40155	00275	11	40260	00160	FINAL OUTPUT TO ES
40156	00276	00	00000	00000	
40157	00277	00	00000	00000	
40160	00300	45	00000	30000	EXIT
40161	00301	11	00040	00127	$0 \to I_7$
40162	00302	11	00040	00131	$0 \to I_{11}$
40163	00303	55	00122	00017	$I_2 \times 2^{15} \to I_2$
40164	00304	11	00121	20000	$I_1 \to (A)$
40165	00305	73	00150	00124	$(A)/6 \to I_4$
40166	00306	55	00121	00017	$I_1 \times 2^{15} \to I_1$
40167	00307	11	00064	20000	63 → (A)
40170	00310	73	00124	00125	$(A)/I_4 \to I_5$ AND (Q)
40171	00311	55	10000	00017	$(Q) \times 2^{15} \to (Q)$
40172	00312	11	10000	00132	$I_5 \times 2^{15} \to I_{12}$
40173	00313	11	00107	20000	$n \to (A)$
40174	00314	73	00125	10000	$n/I_5 \to (Q)$, REM → (A)
40175	00315	11	20000	00126	REM → I_6
40176	00316	54	20000	00017	REM $\times 2^{15} \to (A)$
40177	00317	11	20000	00133	$I_6 \times 2^{15} \to I_{13}$

PX 71900-10-(181)

CONTINUOUS MATRIX MULTIPLIER

40200	00320	47	00323	00321	IS (A) = 0 ?
40201	00321	11	00125	00126	YES. $I_5 \rightarrow I_6$
40202	00322	11	00132	00133	$I_{12} \rightarrow I_{13}$
40203	00323	71	00124	00132	NO. $I_4 \times I_{12} \rightarrow (A)$
40204	00324	11	00147	10000	2 DIGIT U-EXTRACTOR $\rightarrow$ (Q)
40205	00325	53	20000	00327	SET NO. OF CARDS TO READ
40206	00326	53	00132	00364	SET NO OF r's TO TRANSFER TO MD
40207	00327	1457	0000	0377	READ PRESET NO. OF CARDS INTO ES
40210	00330	16	00123	00365	$\ell(d_{11}) \rightarrow \ell(d_{ij})$
40211	00331	21	00365	00131	$\ell(d_{11}) + I_{11} \rightarrow \ell(d_{ij})$
40212	00332	21	00131	00125	$I_{11} + I_5 \rightarrow I_{11}$
40213	00333	11	00004	20000	(00004) $\rightarrow$ (A)
40214	00334	47	00335	00352	IS (A) = 0 ? YES. 352 $\rightarrow$ (PAK)
40215	00335	11	00064	00127	NO. 63 $\rightarrow I_7$
40216	00336	16	00143	00350	$\ell(r_1) \rightarrow \ell(r_{I_6+1})$
40217	00337	21	00350	00126	$\ell(r_1) + I_6 \rightarrow \ell(r_{I_6+1})$
40220	00340	31	00101	00000	$k_2 \rightarrow (R)$
40221	00341	34	00107	00017	$(k_2 - m) \times 2^{15} \rightarrow (R)$
40222	00342	11	20000	00124	$(R) \rightarrow I_4$
40223	00343	11	00146	10000	4 DIGIT U-EXTRACTOR $\rightarrow$ (Q)
40224	00344	53	20000	00347	SET NO. OF r's TO BE ZERO
40225	00345	21	00124	00133	$I_4 \times I_{13} \rightarrow I_4$
40226	00346	53	20000	00364	SET NO. OF r's TO TRANSFER TO MD
40227	00347	75	10000	00351	SET PROPER
40230	00350	11	00040	30000	r's = 0
40231	00351	11	00126	00125	$I_6 \rightarrow I_5$
40232	00352	11	00040	00130	$0 \rightarrow I_{10}$
40233	00353	15	00143	00360	$\ell(\delta_{11}) \rightarrow \ell(\delta_{ij})$
40234	00354	21	00360	00130	$\ell(\delta_{11}) + I_{10} \rightarrow \ell(\delta_{ij})$
40235	00355	16	00143	00360	$\ell(r_1) \rightarrow \ell(r_j)$
40236	00356	11	00125	20000	$I_5 \rightarrow (A)$
40237	00357	36	00074	00124	$I_5 - 1 \rightarrow I_4$

CONTINUOUS MATRIX MULTIPLIER

40240	00360	11	30000	30000	$\delta_{ij} \rightarrow r_j$
40241	00361	21	00360	00074	$\ell(r_j)+1 \rightarrow \ell(r_j)$
40242	00362	35	00121	00360	$\ell(\delta_{ij}) + I_1 \rightarrow \ell(\delta_{ij})$
40243	00363	41	00124	00360	TEST INDEX
40244	00364	75	30000	00366	SEND ALL
40245	00365	11	01177	30000	r_j's $\rightarrow d_{ij}$
40246	00366	21	00365	00101	$\ell(d_{ij}) + k_2 \rightarrow \ell(d_{ij})$
40247	00367	21	00130	00073	$I_{10} + 1 \times 2^{15} \rightarrow I_{10}$
40250	00370	42	00122	00353	TEST INDEX
40251	00371	11	00127	20000	$I_7 \rightarrow (A)$
40252	00372	47	00300	00327	TEST FOR MORE CARD READS
40253	00373	00	00000	00000	
40254	00374	00	00000	00000	
40255	00375	00	00000	00000	
40256	00376	00	00000	00000	
40257		00	00000	00000	
40260	00160	00	00000	00000	
40261	00161	00	00000	00000	
40262	00162	00	00000	00000	
40263	00163	00	00000	00000	
40264	00164	00	00000	00000	
40265	00165	00	00000	00000	
40266	00166	00	00000	00000	
40267	00167	00	00000	00000	
40270	00170	56	30000	00171	MS-3
40271	00171	16	00143	00203	$\ell(r_i) \rightarrow \ell(r_j)$
40272	00172	21	00203	00106	$\ell(r_1) + m \rightarrow \ell(r_{m+1})$
40273	00173	31	00100	00000	$k_1 \rightarrow (R)$
40274	00174	34	00106	00017	$(k_1 - m) \times 2^{15} \rightarrow (A)$
40275	00175	11	00146	10000	4 DIGIT U-EXTRACTOR $\rightarrow$ (Q)
40276	00176	53	20000	00202	SET NO. OF r's TO BE ZERO
40277	00177	53	00104	00210	SET NO. OF ROWS OF B_c

CONTINUOUS MATRIX MULTIPLIER

40300	00200	53 00104 00216	SET NO. OF COLUMNS OF A_r
40301	00201	53 00103 00231	SET NO. OF ROWS OF C_c
40302	00202	75 10000 00204	SET PROPER
40303	00203	11 00040 30000	$r_i = 0$
40304	00204	15 00141 00211	$\ell(b_{11}) \to \ell(b_{jk})$
40305	00205	16 00142 00232	$\ell(c_{11}) \to \ell(c_{jk})$
40306	00206	11 00110 20000	$s \to (A)$
40307	00207	36 00074 00121	$s - 1 \to I_1$
40310	00210	⌈75 30000 00212	SEND k^{th} COLUMN OF
40311	00211	11 30000 00677	B_C TO ES
40312	00212	15 00140 00217	$\ell(a_{11}) \to \ell(a_{ij})$
40313	00213	16 00143 00225	$\ell(r_1) \to \ell(r_j)$
40314	00214	11 00106 20000	$m \to (A)$
40315	00215	36 00074 00122	$m - 1 \to I_2$
40316	00216	⌈75 30000 00220	SEND i^{th} ROW OF
40317	00217	11 30000 00377	A_r TO ES
40320	00220	11 00101 20000	$k_2 \to (A)$
40321	00221	36 00074 00123	$k_2 - 1 \to I_3$
40322	00222	11 00040 20000	$0 \to (A)$
40323	00223	⌈1423 4377 4677	$(R) + \alpha_j \beta_j \to (R)$
40324	00224	1440 0123 0223⌋	IS $k_1 < k_2 - 1$? YES. 223 → (PAK)
40325	00225	11 20000 30000	NO $(R) \to r_j$
40326	00226	21 00217 00104	$\ell(a_{ij}) + k_2 \to \ell(a_{ij})$
40327	00227	21 00225 00074	$\ell(r_i) + 1 \to \ell(r_i)$
40330	00230	41 00122 00216⌋	TEST INDEX
40331	00231	75 30000 00233	SEND k^{th} COLUMN OF
40332	00232	11 01177 30000	C_C TO MD
40333	00233	21 00232 00100	$\ell(c_{ik}) + k_1 \to \ell(c_{ik})$
40334	00234	21 00211 00104	$\ell(b_{jk}) + k_2 \to \ell(b_{jk})$
40335	00235	41 00121 00210⌋	TEST INDEX
40336	00236	56 30000 00237	MS-3
40337	00237	45 00000 00240	240 → (PAK)

CONTINUOUS MATRIX MULTIPLIER

40340	00240	11	00114	20000	$\rho_4 \rightarrow$ (A)
40341	00241	47	00244	00246	IS $\rho_4 = 0$?
40342	00242	00	00000	00000	
40343	00243	00	00000	00000	
40344	00244	75	30120	00160	NO. INTERMEDIATE OUTPUT
40345	00245	11	40500	00160	AND NEW SETUP TO ES
40346	00246	56	10000	00247	YES. MS-1
40347	00247	11	00113	20000	$\rho_3 \rightarrow$ (A)
40350	00250	47	00251	00257	IS $\rho_3 = 0$?
40351	00251	37	00301	00302	NO. $C_c' \rightarrow$ A
40352	00252	37	00350	00351	PUNCH C_c'
40353	00253	45	10000	00255	MJ-1. IF ON, 255 $\rightarrow$ (PAK)
40354	00254	45	00000	00264	264 $\rightarrow$ (PAK)
40355	00255	37	00340	00341	PUNCH C_c
40356	00256	45	00000	00264	264 $\rightarrow$ (PAK)
40357	00257	37	00340	00341	PUNCH C_c
40360	00260	45	10000	00262	MJ-1. IF ON, 262 $\rightarrow$ (PAK)
40361	00261	45	00000	00264	264 $\rightarrow$ (PAK)
40362	00262	37	00301	00302	$C_c' \rightarrow$ A
40363	00263	37	00350	00351	PUNCH C_c'
40364	00264	37	73374	73374	ADVANCE ALL CARDS PROCESSED
40365	00265	56	00000	40365	STOP
40366	00266	00	00000	00000	
40367	00267	00	00000	00000	
40370	00270	00	00000	00000	
40371	00271	00	00000	00000	
40372	00272	00	00000	00000	
40373	00273	00	00000	00000	
40374	00274	00	00000	00000	
40375	00275	00	00000	00000	
40376	00276	00	00000	00000	
40377	00277	00	00000	00000	

PX 71900-10-(181)

CONTINUOUS MATRIX MULTIPLIER

40400	00300	37	76000	76002	ALARM EXIT
40401	00301	45	00000	30000	EXIT
40402	00302	16	00143	00312	$\ell(r_1) \to \ell(r_j)$
40403	00303	21	00312	00110	$\ell(r_1) + s \to \ell(r_{s+1})$
40404	00304	31	00102	00000	$k_3 \to (R)$
40405	00305	34	00110	00017	$(k_3 - s) \times 2^{15} \to (R)$
40406	00306	11	00146	10000	4 DIGIT U-EXTRACTOR $\to$ (Q)
40407	00307	53	20000	00311	SET NO. OF r's TO BE ZERO
40410	00310	53	00105	00325	SET NO. OF ROWS OF C_c'
40411	00311	75	10000	00313	SET PROPER
40412	00312	11	00040	30000	r's $= 0$
40413	00313	11	00040	00122	$0 \to I_2$
40414	00314	15	00142	00321	$\ell(c_{11}) \to \ell(c_{ij})$
40415	00315	16	00140	00326	$\ell(a_{11}) \to \ell(a_{ij})$
40416	00316	11	00110	20000	$s \to (A)$
40417	00317	36	00074	00121	$s - 1 \to I_1$
40420	00320	16	00143	00321	$\ell(r_1) \to \ell(r_j)$
40421	00321	11	30000	30000	$c_{ij} \to r_j$
40422	00322	21	00321	00074	$\ell(r_j) + 1 \to \ell(r_j)$
40423	00323	35	00103	00321	$\ell(c_{ij}) + k_1 \to \ell(c_{ij})$
40424	00324	41	00121	00321	TEST INDEX
40425	00325	75	30000	00327	SEND i^{th} COLUMN OF
40426	00326	11	01177	30000	C_c' TO MD
40427	00327	21	00326	00102	$\ell(a_{ij}) + k_3 \to \ell(a_{ij})$
40430	00330	21	00122	00074	$I_2 + 1 \to I_2$
40431	00331	54	20000	00017	$(A) \times 2^{15} \to (A)$
40432	00332	15	00142	00321	$\ell(c_{11}) \to \ell(c_{ij})$
40433	00333	35	00321	00321	$\ell(c_{11}) + I_2 \to \ell(c_{ij})$
40434	00334	11	00122	20000	$I_2 \to (A)$
40435	00335	42	00106	00316	TEST INDEX
40436	00336	45	00000	00301	TO EXIT
40437	00337	00	00000	00000	

CONTINUOUS MATRIX MULTIPLIER

40440	00340	45 00000 30000	EXIT
40441	00341	71 00100 00110	$k_1 \times n \to (A)$
40442	00342	73 00150 20000	$(A)/6 \to (A)$
40443	00343	16 00142 00365	$\ell(c_{11}) \to \ell(d_{ij})$
40444	00344	37 00360 00361	PUNCH C_c
40445	00345	17 00000 00077	BLANK CARD IN OUTPUT
40446	00346	45 00000 00340	TO EXIT
40447	00347	00 00000 00000	
40450	00350	45 00000 30000	EXIT
40451	00351	71 00102 00106	$m \times k_3 \to (A)$
40452	00352	73 00150 20000	$(A)/6 \to (A)$
40453	00353	16 00140 00365	$\ell(a_{11}) \to \ell(d_{ij})$
40454	00354	37 00360 00361	PUNCH C_c'
40455	00355	17 00000 00077	BLANK CARD IN OUTPUT
40456	00356	45 00000 00350	TO EXIT
40457	00357	00 00000 00000	
40460	00360	45 00000 30000	EXIT
40461	00361	42 00064 00371	IS 63 > (A)?. YES, 371 → (PAK)
40462	00362	73 00064 00122	NO. (A)/63 → I_2, REM → (A)
40463	00363	11 20000 00121	(A) → I_1
40464	00364	23 00122 00074	$I_2 - 1 \to I_2$
40465	00365	⌈1456 0773 0000	PUNCH 63 CARDS STARTING AT $\ell(d_{ij})$
40466	00366	21 00365 00151	$\ell(d_{ij}) + 378 \to \ell(d_{ij})$
40467	00367	41 00122 00365⌋	TEST INDEX
40470	00370	11 00121 20000	$I_2 \to (A)$
40471	00371	16 00365 00375	$\ell(d_{ij}) \to \ell(d_{ij})$
40472	00372	54 20000 00017	$(A) \times 2^{15} \to (A)$
40473	00373	11 00147 10000	2 DIGIT U-EXTRACTOR → (Q)
40474	00374	53 20000 00375	SET NO. CARDS TO PUNCH
40475	00375	1456 0003 0000	PUNCH CARDS
40476	00376	45 00000 00360	TO EXI
40477		00 00000 00000	

PX 71900-10-(181)

CONTINUOUS MATRIX MULTIPLIER

40500	00160	56	20000	00161	MS-2
40501	00161	11	00113	20000	$\rho_3 \rightarrow (A)$
40502	00162	47	00163	00210	IS $\rho_3 = 0$? YES, 210 $\rightarrow$ (PAK)
40503	00163	45	10000	00167	NO, MJ-1, IF ON 167 $\rightarrow$ (PAK)
40504	00164	45	30000	00173	MJ-3, IF ON 173 $\rightarrow$ (PAK)
40505	00165	45	20000	00177	MJ-2, IF ON 177 $\rightarrow$ (PAK)
40506	00166	45	00000	00200	200 $\rightarrow$ (PAK)
40507	00167	45	20000	00177	MJ-2, IF ON 177 $\rightarrow$ (PAK)
40510	00170	37	00301	00302	$C_c' \rightarrow A$
40511	00171	37	00350	00351	PUNCH C_c'
40512	00172	45	00000	00200	200 $\rightarrow$ (PAK)
40513	00173	37	00301	00302	$C_c' \rightarrow A$
40514	00174	37	00350	00351	PUNCH C_c'
40515	00175	45	20000	00177	MJ-2, IF ON 177 $\rightarrow$ (PAK)
40516	00176	45	00000	00200	200 $\rightarrow$ (PAK)
40517	00177	37	00340	00341	PUNCH C_c
40520	00200	71	00103	00110	$k_1 \cdot s \times 2^{15} \rightarrow (A)$
40521	00201	11	00146	10000	4 DIGIT U-EXTRACTOR $\rightarrow$ (Q)
40522	00202	53	20000	00203	SET NO. OF ELEMENTS OF C_c
40523	00203	75	30000	00205	SEND
40524	00204	11	60210	40630	$C_c \rightarrow A$
40525	00205	11	00106	00107	$m \rightarrow n$
40526	00206	11	00110	00106	$s \rightarrow m$
40527	00207	45	00000	00225	225 $\rightarrow$ (PAK)
40530	00210	37	00301	00302	$C_c' \rightarrow A$
40531	00211	45	10000	00215	MJ-1, IF ON 215 $\rightarrow$ (PAK)
40532	00212	45	20000	00220	MJ-2, IF ON 220 $\rightarrow$ (PAK)
40533	00213	45	30000	00223	MJ-3, IF ON 223 $\rightarrow$ (PAK)
40534	00214	45	00000	00224	224 $\rightarrow$ (PAK)
40535	00215	45	20000	00223	MJ-2, IF ON 223 $\rightarrow$ (PAK)
40536	00216	37	00340	00341	PUNCH C_c
40537	00217	45	00000	00224	224 $\rightarrow$ (PAK)

CONTINUOUS MATRIX MULTIPLIER

40540	00220	37	00340	00341	PUNCH C_c
40541	00221	45	30000	00223	MJ-3, IF ON 223 → (PAK)
40542	00222	45	00000	00224	224 → (PAK)
40543	00223	37	00350	00351	PUNCH C_c'
40544	00224	11	00110	00107	$s \rightarrow n$
40545	00225	56	30000	00226	MS-3
40546	00226	1457	1010	0110	READ NEW PARAMETER AND FLAGS
40547	00227	45	00000	00230	230 → (PAK)
40550	00230	11	00041	00121	SET INDEX
40551	00231	15	00144	00234	$\ell(m) \rightarrow \ell(m, n, s)$
40552	00232	16	00144	00241	$\ell(k_1) \rightarrow \ell(k_i)$
40553	00233	16	00145	00243	$\ell(k_1 \times 2^{15}) \rightarrow \ell(k_i \times 2^{15})$
40554	00234	11	30000	20000	$m, n,$ OR $s \rightarrow$ (A)
40555	00235	73	00150	10000	(A)/6 → (Q), REM → (A)
40556	00236	47	00237	00240	IS REM = 0?
40557	00237	21	10000	00074	NO. (Q)+1 → (Q)
40560	00240	71	10000	00150	YES. (Q) × 6 → (A)
40561	00241	11	20000	30000	(R) → k_i
40562	00242	54	20000	00017	(A) $\times 2^{15} \rightarrow$ (A)
40563	00243	11	20000	30000	(R) → $k_i \times 2^{15}$
40564	00244	21	00234	00073	$\ell(m, n, s) + 1 \rightarrow \ell(m, n, s)$
40565	00245	21	00241	00074	$\ell(k_i) + 1 \rightarrow \ell(k_i)$
40566	00246	21	00243	00074	$\ell(k_i \times 2^{15}) + 1 \rightarrow \ell(k_i \times 2^{15})$
40567	00247	41	00121	00234	TEST INDEX
40570	00250	71	00101	00110	$k_2 \times s \rightarrow$ (A)
40571	00251	42	00152	00253	IS 3961 $> k_2 \times s$?
40572	00252	37	76000	76001	NO. ALARM EXIT
40573	00253	71	00100	00110	YES. $k_1 \times s \rightarrow$ (A)
40574	00254	42	00152	00256	IS 3961 $> k_1 \times s$?
40575	00255	37	76000	76001	NO. ALARM EXIT
40576	00256	71	00102	00106	YES. $m \times k_3 \rightarrow$ (A)
40577	00257	42	00152	00261	IS 3961 $> m \times k_3$?

PX 71900-10-(181)

CONTINUOUS MATRIX MULTIPLIER

40600	00260	37	76000	76001	NO. ALARM EXIT
40601	00261	11	00110	20000	YES. 4 → (A)
40602	00262	42	00153	00270	IS 193 > 4 ? YES, 270 → (PAK)
40603	00263	37	76000	76001	NO. ALARM EXIT
40604	00264	00	00000	00000	
40605	00265	00	00000	00000	
40606	00266	00	00000	00000	
40607	00267	00	00000	00000	
40610	00270	75	30120	00255	NEW MATRIX CARD
40611	00271	11	40135	00255	READ TO ES
40612	00272	00	00000	00000	
40613	00273	00	00000	00000	
40614	00274	00	00000	00000	
40615	00275	00	00000	00000	
40616	00276	00	00000	00000	
40617	00277	00	00000	00000	
40620		37	70563	70567	BLOTTOS
40621		00	00100	00377	ES AND
40622		00	40000	40637	MD.
40623		56	00000	40623	STOP
40624		00	00000	00000	
40625		00	00000	00000	
40626		00	00000	00000	
40627		00	00000	00000	

CONTINUOUS MATRIX MULTIPLIER USING SINGLE OR MULTI-PRECISION ARITHMETIC

This routine multiplies two matrices, take this product or its transpose times a third matrix, the resulting product or its transpose times a fourth, etc. All input and output data is by means of cards with five floating decimal numbers per card (see IC-004 for card format). The elements of any one of the matrices may be punched on cards with the elements of a row or a column in consecutive order with or without each row or column starting at the beginning of a new card.* The input assumes all matrices are punched on the cards by columns in packed form (i.e., no spaces are left on the cards between the last element of one column and the first element of the next). However, the routine will handle matrices punched by rows and/or in unpacked form. Flags punched as numbers on cards are used to indicate the form of the matrices on the cards. Flags are also used to indicate the use of single or multi-precision arithmetic, the use of the transpose of any of the products, and the continuation of the multiplication. Three parameters are necessary to furnish the routine with the size of the matrices. The flags are denoted by the symbols P_o to P_6 and the parameters by m, n, and s.

* The stop read indicator (a 12-punch in column 1) must be on the last card for each matrix.

The flags and parameters are punched as the mantissas on the parameter cards and are read into the memory as fixed point integers (see I C - 011). Two parameters, m and n, and three flags, P_o, P_1, and P_2, are on the first of two parameter cards while s and P_3 to P_6 are on the second card. The parameters and flags are defined as follows and appear on the cards in this order:

FIRST CARD

m number of rows of the first matrix

n number of columns of the first matrix (also, number of rows of the second)

P_o =0 for single-precision arithmetic
≠0 for multi-precision arithmetic

P_1 =0 for first matrix punched packed
≠0 for first matrix punched unpacked

P_2 =0 for first matrix punched by columns
≠0 for first matrix punched by rows

SECOND CARD

s number of columns of the second matrix

P_3 =0 for second matrix punched packed
≠0 for second matrix punched unpacked

P_4 =0 for second matrix punched by columns
≠0 for second matrix punched by rows

P_5 =0 for product to be used
≠0 for transpose of product to be used

P_6 =0 for no further multiplication
≠0 for continued multiplication

Examination of this card format reveals that before any multiplication both parameter cards must be used while only the second card is necessary before each matrix in a continued product.

To illustrate the use of the parameters and flags, suppose the product, ABC, of three matrices is wanted using single-precision arithmetic where the matrices have the following properties:

A: m by n, punched packed by rows

B: n by s, punched unpacked by columns

C: s by t, punched unpacked by rows

The multiplication proceeds from the left so AB is the first product formed then this product multiplies C. The first input cards contains as mantissas m,n,0,0, and non-zero. The second input card contains as mantissas s, non-zero, 0, 0, and non-zero. The cards for A and B are the next input cards then one parameter card followed by the cards for C. The input card before C contains as mantissas t, non-zero, non-zero, 0 and 0. No blank cards appear in the input. The last card of each matrix contains a 12-punch in column one to indicate the end of that matrix.

Output of the intermediate products and the final product is controlled by the MJ and MS switches. The intermediate output is preceded by the only MS-2 and followed by the only MS-3. Setting (PAK) = 00176 at MS-3 and starting will give additional intermediate output if the desired MJ switches are set. The final output is preceded by the only MS-1 and followed by MS-0. Starting at MS-0 will give additional final output if the desired MJ switches are set. If no MJ switches are set when the final product is formed, the typewriter prints "SET MJ-S FOR OUTPUT" before stopping at MS-0.

Setting the MJ's and starting will give the final output. All output is by columns with the column and row indices of the first element on each card indicated as the card number (see IC-004). The form of the output as controlled by the MJ switches is as follows:

M J's	Output for $P_5=0$	Output for $P_5\neq 0$
none	none	none
1	product packed	transpose packed
2	product unpacked	transpose unpacked
1 and 3	transpose packed	product packed
2 and 3	transpose unpacked	product unpacked
1 and 2	same as 1	same as 1
1, 2, and 3	same as 1 and 3	same as 1 and 3
3	none	none

This routine, the working storage, the matrix storage, and the service routines use the entire ES and most of MD storage. The storage is allotted as follows:

00000 - 00070	temporaries
00071 - 00147	constants
00150 - 00457	program
00460 - 01777	working storage
40000 - 41377	program
41400 - 41477	available
41500 - 54637	first matrix
54640 - 67777	second matrix and product
70000 - 77777	service routines, IC - 004, and IC - 011.

Tests in the routine will give an alarm exit if the size of a matrix is too large for the storage space provided. The following restrictions are imposed on the size of the matrices:

1. No dimension greater than decimal 120.
2. The number of elements in any matrix not greater than 2860 (this includes the zeros in an unpacked form.)

Thus, the largest square matrix that can be handled is a 53 by 53 packed or a 52 by 52 unpacked since an unpacked form would be handled on cards as a 52 by 55 containing 2860 elements including the zeros. (Note: even though the matrix is unpacked and the storage limitations require consideration of this, the input dimensions on the parameter cards are the packed dimensions.)

A check sum, equal to 012345670123, is printed at the beginning of the program after reading the paper tape. This represents the sum of this routine, IC-004, and IC-011, all of which are loaded by paper tape. A programmed Blotto is included in the code. Setting (PAK) =41,370 and starting gives a Blotto of 00000-00457 and 40000-41377 which includes the temporaries, constants, and program in E.S. and entire program on MD.

The program starts with (PAK) =40,000. Several continuous multiplications may be done by resetting (PAK) =40,000 at MS-0. However, a blank card must be placed between the cards for the separate continuous multiplications unless the reading hopper is loaded when PAK is reset. The check sum obtained when PAK is reset is not the same as that obtained after reading the paper tape.

CONTINUOUS MATRIX MULTIPLIER

40000		45	00000	40001	
40001		31	40000	00000	CHECK
40002		75	21377	41374	SUM
40003		32	40001	00000	PROGRAM.
40004		11	20000	10000	
40005		17	00000	72433	PRIME BULL.
40006		75	30370	00150	PROGRAM
40007		11	40010	00070	TO ES.
40010	00070	00	00000	00000	
40011	00071	00	00000	00110	DEC. 72.
40012	00072	40	00000	00000	SIGN MASK.
40013	00073	00	00000	00003	
40014	00074	00	00000	00001	
40015	00075	77	77777	70000	MANTISSA AND EXPONENT MASK.
40016	00076	77	77777	77777	NEGATIVE ZERO.
40017	00077	00	00000	00152	DEC. 106.
40020	00100	00	00000	00045	DEC. 37.
40021	00101	00	00000	00042	DEC. 34.
40022	00102	00	00000	04000	DEC. 2048.
40023	00103	77	77777	74000	DEC.-2047.
40024	00104	00	00000	00072	DEC. 58.
40025	00105	00	00000	00021	DEC. 17.
40026	00106	00	07777	00000	4-DIGIT U-EXTRACTOR.
40027	00107	00	41500	41500	START OF A.
40030	00110	00	54640	54640	START OF B.
40031	00111	00	00460	00460	START OF
40032	00112	00	01040	01040	ES WORKING
40033	00113	00	01420	01420	STORAGES.
40034	00114	00	00002	00000	
40035	00115	00	00000	77777	V-EXTRACTOR.
40036	00116	00	00000	00002	
40037	00117	47	45242	00145	FLEX CODE

CONTINUOUS MATRIX MULTIPLIER

```
40040   00120   07 32562 40404          FOR--
40041   00121   03 12040 33426          SET MJ-S
40042   00122   15 34010 45701          FOR OUTPUT.
40043   00123   00 00000 00013      DEC. 11.
40044   00124   00 00000 00007
40045   00125   00 00000 00005
40046   00126   00 00000 00004
40047   00127   61 00000 00130      ZERO PRINT ORDER.
40050   00130   00 00000 00037      FLEX
40051   00131   00 00000 00052
40052   00132   00 00000 00074          CODES
40053   00133   00 00000 00070
40054   00134   00 00000 00064          FOR
40055   00135   00 00000 00062
40056   00136   00 00000 00066          0-7.
40057   00137   00 00000 00072
40060   00140   00 00000 00171      DEC. 121.
40061   00141   00 00000 05455      DEC. 2861.
40062   00142   00 00300 00300      START OF ES STORAGE
40063   00143   00 00000 00442      DEC. 290.
40064   00144   00 00000 00043      DEC. 35.
40065   00145   00 00000 00044      DEC. 36.
40066   00146   00 00000 00235      DEC. 157.
40067   00147   00 00000 00000
40070   00150   11 00123 00001      PRINT OF
40071   00151   61 00000 00100
40072   00152   55 10000 00003          CHECK
40073   00153   51 00124 20000
40074   00154   35 00127 00155          SUM.
40075   00155   30 00000 00000
40076   00156   41 00001 00152
40077   00157   37 72400 72401      READ
```

CONTINUOUS MATRIX MULTIPLIER

```
40100  00160  04 00021 00012        PARAMETER CARDS.
40101  00161  11 00023 00060    STORE
40102  00162  11 00026 00023        PARAMETERS
40103  00163  11 00032 00026        CONSECUTIVELY.
40104  00164  11 00021 20000
40105  00165  42 00140 00167    TEST
40106  00166  37 76000 76001
40107  00167  11 00022 20000        SIZE
40110  00170  42 00140 00172
40111  00171  37 76000 76001        OF
40112  00172  11 00023 20000
40113  00173  42 00140 00175        PARAMETERS.
40114  00174  37 76000 76001
40115  00175  75 30003 00177    DOUBLE
40116  00176  11 00021 00032
40117  00177  75 30003 00201        PARAMETERS.
40120  00200  21 00032 00021
40121  00201  75 30003 00203    SCALE
40122  00202  11 00021 00035
40123  00203  75 20003 00205        PARAMETERS.
40124  00204  55 00035 00020
40125  00205  75 30003 00207    DECREASE
40126  00206  11 00021 00041
40127  00207  75 20003 00211        PARAMETERS BY 1.
40130  00210  23 00041 00074
40131  00211  75 30003 00213    SCALE
40132  00212  11 00021 00052
40133  00213  75 20003 00215        PARAMETERS.
40134  00214  55 00052 00017
40135  00215  75 30003 00217    FIND
40136  00216  11 00021 00044
40137  00217  11 00021 20000        FIRST
```

CONTINUOUS MATRIX MULTIPLIER

40140	00220	73	00125	10000	
40141	00221	47	00222	00224	MUTIPLE
40142	00222	71	10000	00125	
40143	00223	35	00125	00044	OF 5
40144	00224	11	00022	20000	
40145	00225	73	00125	10000	GREATER
40146	00226	47	00227	00231	
40147	00227	71	10000	00125	THEN
40150	00230	35	00125	00045	
40151	00231	11	00023	20000	OR EQUAL
40152	00232	73	00125	10000	
40153	00233	47	00234	00236	TO
40154	00234	71	10000	00125	
40155	00235	35	00125	00046	PARAMETERS.
40156	00236	75	30003	00240	
40157	00237	11	00044	00047	DOUBLE
40160	00240	75	30003	00242	
40161	00241	21	00047	00044	MULTIPLES.
40162	00242	75	30003	00244	
40163	00243	11	00044	00055	SCALE
40164	00244	75	20003	00246	
40165	00245	55	00055	00017	MULTIPLES.
40166	00246	71	00021	00045	
40167	00247	42	00141	00251	
40170	00250	37	76000	76001	TEST
40171	00251	71	00045	00023	
40172	00252	42	00141	00254	SIZES
40173	00253	37	76000	76001	
40174	00254	71	00044	00023	OF
40175	00255	42	00141	00257	
40176	00256	37	76000	76001	MATRICES.
40177	00257	71	00046	00021	

CONTINUOUS MATRIX MULTIPLIER

```
40200  00260  42 00141 00262
40201  00261  37 76000 76001
40202  00262  11 00070 20000   TEST FOR
40203  00263  47 00305 00264       READ OF A.
40204  00264  11 00025 20000   TEST FOR
40205  00265  47 00266 00276       CHANGE IN FORM OF A.
40206  00266  37 72400 72401   READ A
40207  00267  00 41500 05544       ONTO MD.
40210  00270  11 00024 20000   TEST FOR
40211  00271  47 00272 00305       UNPACKING A.
40212  00272  11 00041 00001
40213  00273  11 00107 00002
40214  00274  37 00351 00352   PACK A.
40215  00275  45 00000 00305
40216  00276  11 00021 00004   CHANGE
40217  00277  11 00035 00002
40220  00300  11 00107 00003       A AND
40221  00301  11 00024 20000
40222  00302  47 00303 00304       STORE
40223  00303  11 00044 00004
40224  00304  37 40400 40402       ON MD.
40225  00305  11 00030 20000   TEST FOR
40226  00306  47 00317 00307       CHANGE IN FORM OF B.
40227  00307  37 72400 72401   READ B
40230  00310  00 54640 05544       ONTO MD.
40231  00311  11 00027 20000   TEST FOR
40232  00312  47 00313 00326       UNPACKING B.
40233  00313  11 00043 00001
40234  00314  11 00110 00002
40235  00315  37 00351 00352   PACK B.
40236  00316  45 00000 00326
40237  00317  11 00023 00004   CHANGE
```

CONTINUOUS MATRIX MULTIPLIER

```
40240   00320   11 00037 00002        B AND
40241   00321   11 00110 00003
40242   00322   11 00027 20000        STORE
40243   00323   47 00324 00325
40244   00324   11 00046 00004        ON MD.
40245   00325   37 40400 40402
40246   00326   45 00000 00327
40247   00327   11 00036 00040    TEST FOR
40250   00330   11 00022 20000
40251   00331   42 00021 00333        SPACING OF B.
40252   00332   45 00000 00341
40253   00333   11 00110 00001    SPACE
40254   00334   11 00033 00002
40255   00335   11 00032 00003        COLUMNS
40256   00336   11 00023 00004
40257   00337   37 00371 00372        OF B.
40260   00340   11 00035 00040
40261   00341   75 30060 00343    MULTIPLIER
40262   00342   11 40540 00150        TO ES.
40263   00343   11 00060 20000    TEST FOR
40264   00344   47 00347 00345        ARITHMETIC.
40265   00345   75 30100 00150    SINGLE PRECISION
40266   00346   11 40620 00230        TO ES.
40267   00347   75 30150 00150    MULTI-PRECISION
40270   00350   11 40720 00230        TO ES.
40271   00351   45 00000 30000    EXIT-PACKING SUBROUTINE.
40272   00352   11 00106 10000
40273   00353   53 00036 00363    SET
40274   00354   53 00036 00365
40275   00355   23 00001 00074        INITIAL
40276   00356   15 00002 00364
40277   00357   16 00002 00366        CONDITIONS.
```

CONTINUOUS MATRIX MULTIPLIER

```
40300   00360   31 00056 00001   ADVANCE
40301   00361   35 00364 00364       ADDRESSES.
40302   00362   21 00366 00033
40303   00363   75 30000 00365   UNPACKED
40304   00364   11 30000 01420       ROW TO ES.
40305   00365   75 30000 00367   PACKED
40306   00366   11 01420 30000       ROW TO MD.
40307   00367   41 00001 00360
40310   00370   45 00000 00351
40311   00371   45 00000 30000   EXIT-UNPACKING SUBROUTINE.
40312   00372   11 00002 20000
40313   00373   43 00003 00375   TEST
40314   00374   45 00000 00403
40315   00375   13 00076 00005       FOR
40316   00376   13 00076 00006
40317   00377   15 00001 00005       NEEDED
40320   00400   16 00001 00006
40321   00401   54 00006 00017       UNPACKING.
40322   00402   43 00005 00371
40323   00403   11 00003 00005
40324   00404   55 00005 00017   SET
40325   00405   55 00002 00017
40326   00406   21 00004 00074       LOCATIONS
40327   00407   15 00001 00431
40330   00410   16 00001 00433       OF LAST
40331   00411   71 00002 00004
40332   00412   35 00431 00431       COLUMN
40333   00413   71 00003 00004
40334   00414   35 00433 00433       OR ROW.
40335   00415   11 00106 10000
40336   00416   53 00002 00430   SET
40337   00417   53 00005 00432
```

CONTINUOUS MATRIX MULTIPLIER

40340	00420	23	00005	00002	NUMBER
40341	00421	53	20000	00426	
40342	00422	11	00002	20000	OF ELEMENTS IN
40343	00423	52	00113	20000	
40344	00424	55	20000	00025	ROW OR COLUMN.
40345	00425	16	20000	00427	
40346	00426	75	10000	00430	SET EXTRA
40347	00427	13	00076	30000	ELEMENTS TO ZERO.
40350	00430	75	30000	00432	PACKED ROW OR
40351	00431	11	30000	01420	COLUMN TO ES.
40352	00432	75	30000	00434	UNPACKED ROW OR
40353	00433	11	01420	30000	COLUMN TO MD.
40354	00434	23	00431	00002	RETARD
40355	00435	23	00433	00003	ADDRESSES.
40356	00436	41	00004	00430	
40357	00437	45	00000	00371	
40360	00440	00	00000	00000	
40361	00441	00	00000	00000	
40362	00442	00	00000	00000	
40363	00443	00	00000	00000	
40364	00444	00	00000	00000	
40365	00445	00	00000	00000	
40366	00446	00	00000	00000	
40367	00447	00	00000	00000	
40370	00450	00	00000	00000	
40371	00451	00	00000	00000	
40372	00452	00	00000	00000	
40373	00453	00	00000	00000	
40374	00454	00	00000	00000	
40375	00455	00	00000	00000	
40376	00456	00	00000	00000	
40377	00457	00	00000	00000	

CONTINUOUS MATRIX MULTIPLIER

```
40400  00150  75 30160 30000   EXIT-CHANGE FORM SUBROUTINE.
40401  00151  11 40220 00300   RELOAD ES.
40402  00152  75 30140 00154   LOAD
40403  00153  11 40400 00150       ES.
40404  00154  13 00076 00011
40405  00155  11 00143 20000   FIND
40406  00156  73 00004 00005
40407  00157  71 00004 00005       NUMBER
40410  00160  11 20000 00020
40411  00161  73 00125 00016       OF COLUMNS
40412  00162  11 20000 00015
40413  00163  11 00132 00017       OR ROWS
40414  00164  55 00004 00020
40415  00165  11 00022 20000       PER READ.
40416  00166  73 00005 10000
40417  00167  11 20000 00006
40420  00170  47 00172 00171
40421  00171  11 00005 00006   PRESET
40422  00172  31 00005 00020
40423  00173  11 00106 10000       INDICES
40424  00174  53 20000 00233
40425  00175  11 00115 10000       AND
40426  00176  53 00020 00205
40427  00177  13 00076 00001       TRANSFERS.
40430  00200  11 00015 20000   TEST FOR
40431  00201  47 00202 00204       CARD READ TYPE.
40432  00202  37 00242 00243   READ
40433  00203  45 00000 00206       PRESET
40434  00204  37 72400 72401       NUMBER
40435  00205  00 00300 00000       OF CARDS.
40436  00206  16 00003 00234   PRESET
40437  00207  21 00234 00011       ADDRESSES
```

CONTINUOUS MATRIX MULTIPLIER

```
40440   00210   31 00005 00001         AND
40441   00211   35 00011 00011         INDICES.
40442   00212   11 00001 20000   TEST FOR
40443   00213   47 00214 00220         END OF MATRIX.
40444   00214   11 00106 10000   SET FOR
40445   00215   31 00006 00020         LAST
40446   00216   53 20000 00233         CARD
40447   00217   11 00006 00005         READ.
40450   00220   13 00076 00010   SET COUNTER.
40451   00221   15 00142 00227   PRESET
40452   00222   21 00227 00010         ADDRESSES
40453   00223   16 00113 00227         AND
40454   00224   11 00005 20000         INDICES.
40455   00225   36 00074 00007
40456   00226   75 30002 00230   PUT ROW OR
40457   00227   11 30000 30000         COLUMN
40460   00230   21 00227 00116         ELEMENTS
40461   00231   35 00004 00227         IN ORDER.
40462   00232   41 00007 00226
40463   00233   75 30000 00235   STORE
40464   00234   11 01420 30000         ROW OR
40465   00235   21 00234 00033         COLUMN
40466   00236   21 00010 00114         ON MD.
40467   00237   42 00002 00221
40470   00240   11 00001 20000   TEST FOR
40471   00241   47 00150 00200         LAST CARD READ
40472   00242   45 00000 30000
40473   00243   11 00017 20000   TEST FOR
40474   00244   73 00125 10000         NEED OF
40475   00245   15 00142 00273         ENTIRE
40476   00246   47 00247 00267         LAST CARD.
40477   00247   11 20000 00014
```

CONTINUOUS MATRIX MULTIPLIER

```
40500   00250   54 20000 00020   PRESET
40501   00251   11 00106 10000
40502   00252   53 20000 00260       ADDRESSES.
40503   00253   15 00142 00261
40504   00254   31 00020 00020       TRANSFERS.
40505   00255   35 00261 00261
40506   00256   31 00014 00020       AND INDICES
40507   00257   35 00273 00273
40510   00260   75 30000 00262   PLACE ELEMENTS
40511   00261   11 30000 00300       FROM PREVIOUS READ.
40512   00262   11 00014 20000   TEST FOR NUMBER
40513   00263   42 00015 00267       OF CARDS TO READ.
40514   00264   71 00125 00016   PRESET NUMBER
40515   00265   16 20000 00273       OF NUMBERS.
40516   00266   45 00000 00272
40517   00267   71 00125 00016   PRESET NUMBER
40520   00270   35 00125 20000       OF NUMBERS.
40521   00271   16 20000 00273
40522   00272   37 72400 72401   READ
40523   00273   00 30000 00000       CARDS.
40524   00274   23 00017 00015   PREPARE FOR NEW READ.
40525   00275   45 00000 00242
40526   00276   00 00000 00000
40527   00277   00 00000 00000
40530   00300   00 00000 00000
40531   00301   00 00000 00000
40532   00302   00 00000 00000
40533   00303   00 00000 00000
40534   00304   00 00000 00000
40535   00305   00 00000 00000
40536   00306   00 00000 00000
40537   00307   00 00000 00000
```

CONTINUOUS MATRIX MULTIPLIER

```
40540   00150   45 00000 00151
40541   00151   11 00106 10000   PRESET
40542   00152   53 00035 00211       REPEATS,
40543   00153   53 00036 00160       ADDRESSES,
40544   00154   53 00036 00165       AND
40545   00155   15 00110 00161       INDICES.
40546   00156   16 00110 00212
40547   00157   11 00043 00061
40550   00160   75 30000 00162   ONE COLUMN
40551   00161   11 30000 01040       OF B TO ES.
40552   00162   15 00107 00166   PRESET
40553   00163   16 00113 00205       ADDRESSES
40554   00164   11 00041 00001       AND INDICES.
40555   00165   75 30000 00167   ONE ROW OF
40556   00166   11 30000 00460       A TO ES.
40557   00167   15 00111 00175   PRESET
40560   00170   15 00112 00177       ADDRESSES
40561   00171   11 00042 00002       AND INDICES.
40562   00172   13 00076 00015   ZERO
40563   00173   13 00076 00016       TO RESULT.
40564   00174   75 30002 00176   COMPUTE
40565   00175   11 30000 00011       ONE ELEMENT
40566   00176   75 30002 00200       OF A COLUMN
40567   00177   11 30000 00013       OF C.
40570   00200   37 00231 00232
40571   00201   21 00175 00114   ADVANCE
40572   00202   21 00177 00114       ADDRESSES.
40573   00203   41 00002 00174
40574   00204   75 30002 00206   STORE COMPUTED
40575   00205   11 00015 30000       ELEMENT.
40576   00206   21 00166 00036   ADVANCE
40577   00207   21 00205 00116       ADDRESSES.
```

CONTINUOUS MATRIX MULTIPLIER

```
40600   00210   41 00001 00165
40601   00211   75 30000 00213   ONE COLUMN
40602   00212   11 01420 30000       OF C TO MD.
40603   00213   21 00212 00032   ADVANCE
40604   00214   21 00161 00040       ADDRESSES.
40605   00215   41 00061 00160
40606   00216   75 30310 00150   OUTPUT
40607   00217   11 41070 00150       PROGRAM TO ES.
40610   00220   00 00000 00000
40611   00221   00 00000 00000
40612   00222   00 00000 00000
40613   00223   00 00000 00000
40614   00224   00 00000 00000
40615   00225   00 00000 00000
40616   00226   00 00000 00000
40617   00227   00 00000 00000
40620   00230   37 76000 76002   SINGLE PRECISION
40621   00231   45 00000 30000       ARITHMETIC--SEE ZM-527.
40622   00232   16 00146 00320   SET FOR ACCUMULATE MULTIPLY.
40623   00233   75 30002 00265   STORE
40624   00234   11 00015 00017       SUM.
40625   00235   11 00012 00016   ADD ENTRANCE.
40626   00236   23 10000 10000   CLEAR A AND Q.
40627   00237   16 10000 00257   ERASE SHIFT.
40630   00240   43 00013 00257   Y ZERO TEST.
40631   00241   43 00011 00255   X ZERO TEST.
40632   00242   23 00012 00014   EXPONENT DIFFERENCE.
40633   00243   12 20000 20000
40634   00244   42 00145 00247   36 COMPARSION.
40635   00245   41 00012 00257   Y OR X.
40636   00246   45 00000 00255
40637   00247   16 20000 00257   SET SHIFT.
```

CONTINUOUS MATRIX MULTIPLIER

40640	00250	41	00012	00253	Y OR X.
40641	00251	11	00011	10000	X TO Q.
40642	00252	45	00000	00256	
40643	00253	11	00013	10000	Y TO Q.
40644	00254	75	00001	00257	
40645	00255	11	00014	00016	Y EXPONENT.
40646	00256	11	00013	00011	Y TO X.
40647	00257	54	00011	30000	SHIFT X.
40650	00260	35	10000	00015	
40651	00261	45	00000	00272	TO NORMALIZE.
40652	00262	23	00016	00145	A LEFT
40653	00263	54	00015	00044	NOT SIGNIFICANT.
40654	00264	47	00273	00313	ZERO TEST.
40655	00265	11	00014	20000	MULTIPLY ENTRANCE.
40656	00266	35	00012	20000	ADD EXPONENTS.
40657	00267	36	00144	00016	
40660	00270	71	00013	00011	FORM PRODUCT.
40661	00271	11	20000	00015	TEST
40662	00272	43	20000	00262	EXTENTION.
40663	00273	13	00076	00005	CLEAR FOR SF.
40664	00274	11	00074	10000	POSITIVE FLAG.
40665	00275	46	00276	00277	SIGN TEST.
40666	00276	13	10000	10000	NEGATIVE FLAG.
40667	00277	11	10000	00006	FOR ROUNDING.
40670	00300	74	20000	00005	SCALE FACTOR.
40671	00301	11	20000	00015	
40672	00302	46	00303	00304	ADJUST
40673	00303	13	10000	10000	FOR SIGN.
40674	00304	21	00016	00005	ADJUST FOR SF.
40675	00305	44	00306	00313	
40676	00306	21	00015	00006	ROUND.
40677	00307	43	20000	00313	OVERFLOW TEST.

CONTINUOUS MATRIX MULTIPLIER

```
40700   00310   32 00147 00107   ADJUST
40701   00311   11 20000 00015       FOR
40702   00312   21 00016 00074       OVERFLOW.
40703   00313   11 00015 20000   ZERO
40704   00314   47 00316 00315       TEST.
40705   00315   11 20000 00016   ERASE EXPONENT.
40706   00316   75 30004 00320   SET FOR
40707   00317   11 00015 00011       ADD.
40710   00320   37 00320 00321   TO ADD OR EXIT.
40711   00321   45 00000 00231
40712   00322   00 00000 00000
40713   00323   00 00000 00000
40714   00324   00 00000 00000
40715   00325   00 00000 00000
40716   00326   00 00000 00000
40717   00327   00 00000 00000
40720   00230   37 76000 76002   MULTI-PRECISION
40721   00231   45 00000 30000       ARITHMETIC--SEE CA-006.
40722   00232   75 30002 00246   STORE
40723   00233   11 00015 00017       SUM.
40724   00234   75 10002 00301   ZERO TO
40725   00235   13 00076 00015       MULTIPLY RESULT.
40726   00236   42 00102 00240   TEST EXPONENT.
40727   00237   45 00000 00230   ALARM.
40730   00240   42 00103 00234   TEST EXPONENT.
40731   00241   13 00075 10000   STORE
40732   00242   53 20000 00016       EXPONENT.
40733   00243   11 00072 10000   STORE
40734   00244   53 00015 00016       SIGN.
40735   00245   45 00000 30000   TO ADD OR EXIT.
40736   00246   71 00012 00013   MULTIPLY ENTRANCE.
40737   00247   72 00011 00014   FORM
```

CONTINUOUS MATRIX MULTIPLIER

40740	00250	54	20000	00044	LOW
40741	00251	11	20000	20000	ORDER
40742	00252	54	20000	00001	PRODUCT.
40743	00253	72	00011	00013	ADD HIGHER ORDER.
40744	00254	37	00300	00271	NORMALIZE.
40745	00255	23	00005	00074	
40746	00256	31	00012	00030	STORE
40747	00257	11	20000	00003	
40750	00260	54	00003	00060	EXPONENTS.
40751	00261	31	00014	00030	
40752	00262	11	20000	00010	
40753	00263	54	00010	00060	
40754	00264	37	00264	00265	
40755	00265	21	00005	00003	CORRECT
40756	00266	35	00010	20000	EXPONENT.
40757	00267	37	00245	00236	TEST, STORE EXP, SIGN.
40760	00270	45	00000	00301	TO ADD.
40761	00271	47	00272	00234	ZERO TEST.
40762	00272	13	00076	00005	STORE ZERO.
40763	00273	74	20000	00005	NORMALIZE
40764	00274	11	20000	00015	AND
40765	00275	54	20000	00043	STORE
40766	00276	11	20000	00016	RESULT.
40767	00277	23	00005	00101	
40770	00300	45	00000	30000	
40771	00301	75	30004	00303	ADD EXTRANCE.
40772	00302	11	00015	00011	
40773	00303	11	00013	20000	
40774	00304	47	00305	00231	ZERO TEST.
40775	00305	11	00011	20000	
40776	00306	47	00311	00307	ZERO TEST.
40777	00307	75	30002	00231	ZERO TO

CONTINUOUS MATRIX MULTIPLIER

```
41000  00310  11 00013 00015          RESULT.
41001  00311  37 00264 00256      STORE EXPONENTS.
41002  00312  11 00003 20000      SUBTRACT
41003  00313  36 00010 20000          EXPONENTS.
41004  00314  46 00315 00322      NEGATIVE TEST.
41005  00315  11 00013 00005
41006  00316  11 00014 00006      STORE
41007  00317  11 00010 00003
41010  00320  75 30002 00326          LARGER NUMBER.
41011  00321  11 00011 00007
41012  00322  11 00011 00005      STORE
41013  00323  11 00012 00006
41014  00324  11 00013 00007          LARGER NUMBER.
41015  00325  11 00014 00010
41016  00326  12 20000 20000      STORE
41017  00327  13 20000 00015          EXPONENT DIFFERENCE.
41020  00330  42 00101 00341      34 COMPARISON.
41021  00331  42 00104 00334      58 COMPARISON.
41022  00332  75 30002 00231      LARGER
41023  00333  11 00005 00015          TO RESULT.
41024  00334  21 00015 00077
41025  00335  16 20000 00347      SET
41026  00336  13 00076 00004
41027  00337  11 00007 00010          SHIFT.
41030  00340  45 00000 00347
41031  00341  21 00015 00101      SET
41032  00342  16 20000 00344
41033  00343  11 00074 00004          SHIFT.
41034  00344  55 00004 30000      SHIFT CORRECTION.
41035  00345  35 00100 20000      SET
41036  00346  16 20000 00347          SHIFT.
41037  00347  54 00010 30000      SHIFT MANTISSA.
```

CONTINUOUS MATRIX MULTIPLIER

```
41040   00350   54 00006 00107
41041   00351   54 00005 20042   SHIFT
41042   00352   35 00006 20000       MANTISSA
41043   00353   72 00004 00007       AND
41044   00354   35 00010 20000       ADD.
41045   00355   47 00360 00356
41046   00356   75 10002 00231   ZERO TO
41047   00357   13 00076 00015       RESULT.
41050   00360   37 00300 00272   NORMALIZE.
41051   00361   42 00073 00364   EXPONENT
41052   00362   42 00105 00356       TESTS.
41053   00363   36 00071 20000   CORRECT
41054   00364   35 00003 00003       EXPONENT.
41055   00365   42 00102 00367   EXPONENT TEST.
41056   00366   45 00000 00230
41057   00367   42 00103 00356   EXPONENT TEST.
41060   00370   37 00245 00241   STORE EXPONENT, SIGN.
41061   00371   45 00000 00231
41062   00372   00 00000 00000
41063   00373   00 00000 00000
41064   00374   00 00000 00000
41065   00375   00 00000 00000
41066   00376   00 00000 00000
41067   00377   00 00000 00000
41070   00150   45 00000 00151
41071   00151   11 00026 20000   TEST
41072   00152   47 00175 00153       CONTINUOUS FLAG.
41073   00153   56 10000 00154
41074   00154   45 10000 00173   TEST
41075   00155   45 20000 00173       MJ SWITCHES.
41076   00156   11 00073 00001
41077   00157   11 00071 20000   TYPES--
```

CONTINUOUS MATRIX MULTIPLIER

```
41100   00160   32 00124 00017
41101   00161   15 20000 00162          SET
41102   00162   11 30000 10000
41103   00163   11 00125 00002          MJ-S
41104   00164   61 00000 10000
41105   00165   55 10000 00006          FOR
41106   00166   41 00002 00164
41107   00167   31 00074 00017          OUTPUT.
41110   00170   35 00162 00162
41111   00171   41 00001 00162
41112   00172   56 00000 00154
41113   00173   37 00220 00221   OUTPUT.
41114   00174   56 00000 00154
41115   00175   56 20000 00176
41116   00176   45 10000 00201   TEST
41117   00177   45 20000 00201          MJ SWITCHES.
41120   00200   45 00000 00202
41121   00201   37 00220 00221   OUTPUT.
41122   00202   56 30000 00203
41123   00203   11 00031 20000   TEST FOR
41124   00204   47 00210 00205          TRANSPOSE
41125   00205   37 00272 00273          AND SEND
41126   00206   11 00023 00022          TO POSITION.
41127   00207   45 00000 00213
41130   00210   37 00316 00317   PRODUCT
41131   00211   11 00021 00022          TO POSITION.
41132   00212   11 00023 00021
41133   00213   11 00115 00070
41134   00214   37 72400 72401   READ NEW
41135   00215   04 00026 00005          PARAMETER CARD.
41136   00216   75 30310 00162   LOAD
41137   00217   11 40070 00150          ES.
```

CONTINUOUS MATRIX MULTIPLIER

```
41140   00220   45 00000 30000   EXIT-OUTPUT SUBROUTINE.
41141   00221   11 00031 20000   TEST FOR
41142   00222   47 00263 00223       TRANSPOSE.
41143   00223   45 10000 00245   TEST
41144   00224   45 20000 00226       MJ SWITCHES.
41145   00225   45 00000 00220
41146   00226   45 30000 00236
41147   00227   37 00335 00336   FORM PRODUCT UNPACKED.
41150   00230   15 00054 00234   SET
41151   00231   16 00044 00234       PARAMETERS.
41152   00232   37 70440 70443   PUNCH.
41153   00233   00 41500 00000
41154   00234   00 00000 00000
41155   00235   45 00000 00220
41156   00236   37 00345 00346   FORM TRANSPOSE UNPACKED.
41157   00237   15 00052 00243   SET
41160   00240   16 00046 00243       PARAMETERS.
41161   00241   37 70440 70443   PUNCH.
41162   00242   00 41500 00000
41163   00243   00 00000 00000
41164   00244   45 00000 00220
41165   00245   45 30000 00254
41166   00246   15 00054 00252   SET
41167   00247   16 00021 00252       PARAMETERS.
41170   00250   37 70440 70443   PUNCH PRODUCT PACKED.
41171   00251   00 54640 00000
41172   00252   00 00000 00000
41173   00253   45 00000 00220
41174   00254   37 00272 00273   FORM TRANSPOSE PACKED.
41175   00255   15 00052 00261   SET
41176   00256   16 00023 00261       PARAMETERS.
41177   00257   37 70440 70443   PUNCH.
```

CONTINUOUS MATRIX MULTIPLIER

```
41200  00260  00 41500 00000
41201  00261  00 00000 00000
41202  00262  45 00000 00220
41203  00263  45 10000 00270   TEST
41204  00264  45 20000 00266       MJ SWITCHES.
41205  00265  45 00000 00220
41206  00266  45 30000 00227   PUNCH PRODUCT UNPACKED.
41207  00267  45 00000 00236   PUNCH TRANSPOSE UNPACKED.
41210  00270  45 30000 00246   PUNCH PRODUCT PACKED.
41211  00271  45 00000 00254   PUNCH TRANSPOSE PACKED.
41212  00272  45 00000 30000   EXIT-FORM TRANSPOSE SUBROUTINE.
41213  00273  11 00106 10000   PRESET
41214  00274  53 00037 00310
41215  00275  13 00076 00002       INDICES
41216  00276  16 00107 00311
41217  00277  15 00110 00304       AND
41220  00300  21 00304 00002
41221  00301  11 00043 00001       ADDRESSES.
41222  00302  16 00113 00304
41223  00303  75 30002 00305   ONE ELEMENT
41224  00304  11 30000 30000       OF PRODUCT TO ES.
41225  00305  21 00304 00116   ADVANCE
41226  00306  35 00035 00304       ADDRESSES.
41227  00307  41 00001 00303
41230  00310  75 30000 00312   ONE COLUMN
41231  00311  11 01420 30000       OF TRANSPOSE TO MD.
41232  00312  21 00311 00034   ADVANCE
41233  00313  21 00002 00114       ADDRESSES.
41234  00314  42 00035 00277
41235  00315  45 00000 00272
41236  00316  45 00000 30000   EXIT-STORE PRODUCT SUBROUTINE.
41237  00317  11 00106 10000   PRESET
```

CONTINUOUS MATRIX MULTIPLIER

```
41240  00320  53 00035 00325        REPEATS.
41241  00321  53 00035 00327        INDICES.
41242  00322  11 00043 00001        AND
41243  00323  15 00110 00326        ADDRESSES.
41244  00324  16 00107 00330
41245  00325  75 30000 00327    ONE COLUMN
41246  00326  11 30000 01420        TO ES.
41247  00327  75 30000 00331    ONE COLUMN
41250  00330  11 01420 30000        TO MD.
41251  00331  21 00326 00035    ADVANCE
41252  00332  21 00330 00032        ADDRESSES.
41253  00333  41 00001 00325
41254  00334  45 00000 00316
41255  00335  45 00000 30000    EXIT-UNPACKING PRODUCT SUBROUTINE.
41256  00336  15 00110 00001    PRESET
41257  00337  16 00107 00001        ADDRESSES
41260  00340  11 00032 00002        AND
41261  00341  11 00047 00003        INDICES.
41262  00342  11 00023 00004
41263  00343  37 00371 00372    UNPACK PRODUCT.
41264  00344  45 00000 00335
41265  00345  45 00000 30000    EXIT-UNPACKING TRANSPOSE SUBROUTINE.
41266  00346  37 00272 00273    STORE TRANSPOSE
41267  00347  11 00107 00001    PRESET
41270  00350  11 00034 00002        ADDRESSES
41271  00351  11 00051 00003        AND
41272  00352  11 00021 00004        INDICES.
41273  00353  37 00371 00372    UNPACK PRODUCT.
41274  00354  45 00000 00345
41275  00355  00 00000 00000
41276  00356  00 00000 00000
41277  00357  00 00000 00000
```

CONTINUOUS MATRIX MULTIPLIER

```
41300  00360  00 00000 00000
41301  00361  00 00000 00000
41302  00362  00 00000 00000
41303  00363  00 00000 00000
41304  00364  00 00000 00000
41305  00365  00 00000 00000
41306  00366  00 00000 00000
41307  00367  00 00000 00000
41310  00370  00 00000 00000
41311  00371  45 00000 30000  EXIT-UNPACKING SUBROUTINE.
41312  00372  11 00002 20000
41313  00373  43 00003 00375  TEST
41314  00374  45 00000 00403
41315  00375  13 00076 00005      FOR
41316  00376  13 00076 00006
41317  00377  15 00001 00005      NEEDED
41320  00400  16 00001 00006
41321  00401  54 00006 00017      UNPACKING.
41322  00402  43 00005 00371
41323  00403  11 00003 00005
41324  00404  55 00005 00017  SET
41325  00405  55 00002 00017
41326  00406  21 00004 00074      LOCATIONS
41327  00407  15 00001 00431
41330  00410  16 00001 00433      OF LAST
41331  00411  71 00002 00004
41332  00412  35 00431 00431      COLUMN
41333  00413  71 00003 00004
41334  00414  35 00433 00433      OR ROW.
41335  00415  11 00106 10000
41336  00416  53 00002 00430  SET
41337  00417  53 00005 00432
```

CONTINUOUS MATRIX MULTIPLIER

```
41340  00420  23 00005 00002          NUMBER
41341  00421  53 20000 00426
41342  00422  11 00002 20000          OF ELEMENTS IN
41343  00423  52 00113 20000
41344  00424  55 20000 00025          ROW OR COLUMN.
41345  00425  16 20000 00427
41346  00426  75 10000 00430      SET EXTRA
41347  00427  13 00076 30000          ELEMENTS TO ZERO.
41350  00430  75 30000 00432      PACKED ROW OR
41351  00431  11 30000 01420          COLUMN TO ES.
41352  00432  75 30000 00434      UNPACKED ROW OR
41353  00433  11 01420 30000          COLUMN TO MD.
41354  00434  23 00431 00002      RETARD
41355  00435  23 00433 00003          ADDRESSES.
41356  00436  41 00004 00430
41357  00437  45 00000 00371
41360  00440  00 00000 00000
41361  00441  00 00000 00000
41362  00442  00 00000 00000
41363  00443  00 00000 00000
41364  00444  00 00000 00000
41365  00445  00 00000 00000
41366  00446  00 00000 00000
41367  00447  27 64146 06130      FOR CONVENIENT CHECK SUM.
41370  00450  37 77763 77767      BLOTTOS
41371  00451  00 00000 00457          ES
41372  00452  00 40000 41377          AND
41373  00453  56 00000 41373          MD.
41374  00454  75 20550 41376      SUMS
41375  00455  32 70440 00000          IC-004
41376  00456  75 20352 40004          AND
41377  00457  32 72400 00000          IC-011.
```

CONTINUOUS MATRIX MULTIPLIER

SEE IC-004 FOR CODE OF 70440-71207.

SEE IC-011 FOR CODE OF 72400-72751 EXCEPT FOR THE FOLLOWING CHANGE WHICH STORES THE FLAG FOR THE STOP READ INDICATOR IN CELL 00001 AFTER ES HAS BEEN RESTORED.

```
72532  01074  11 01255 74001
```

DIVISION San Diego

MODEL ALL

REPORT ZM-527

DATE 8/29/56

TITLE

S P U R

Single precision unpacked rounded floating point package for ERA-1103 Computers

Part I Operation Specifications

PREPARED BY I. Barton

GROUP Digital Computing Lab.

REFERENCE

CHECKED BY C. J. Swift
Donn Parker
Dick Bielsker

APPROVED BY [signature]

NO. OF PAGES 22

NO. OF DIAGRAMS 0

REVISIONS

NO.	DATE	BY	CHANGE	PAGES AFFECTED

GO 241A

PX 71900-10-(183)

SINGLE PRECISION UNPACKED ROUNDED FLOATING POINT PACKAGE

I Brief Specifications

Occupies

ES	00000, 00001
	01500 to 01777 inclusive
MD	76000 to 77777 some parts open
	74000 to 75777 (E. S. image)

Registers

P	01764, 01765
B	01766, 01767
C	01770, 01771
S	01772, 01773
R	01774, 01775
b_1	01776,
b_2	01777

Constants used and available

01750 00 00000 00000

01751 00 00000 00001

01752 00 00000 00002

01753 00 00002 00000

01754 00 00000 00043

01755 00 00000 00044

01756 00 00000 00177

01757 00 00000 01777

Activation

37 76000 76001 (stored package to E.S.)

17 00000 77413 (initially prime cards)

Two address commands

14 CC XXXX YYYY

Addition	(CC)	
	00	Y + X to R
	04	R + Y + X to R
	20	Y + X to R, Y
	24	R + Y + X to R, Y
Subtraction	(CC)	
	01	Y - X to R
	05	R + Y - X to R
	21	Y - X to R, Y
	25	R + Y - X to R,Y
Multiplication	(CC)	
	02	Y · X to R
	06	R + Y · X to R
	22	Y · X to R, Y
	26	R + Y · X to R, Y
Division	(CC)	
	03	Y/X to R
	07	R + Y/X to R
	23	Y/X to R, Y
	27	R + Y/X to R, Y

Threshold jump 17 If X is greater than R jump to Y; otherwise continue with the next instruction, In either case R original will be unchanged in R, S, and B. (CAUTION; due to rounding a jump may occur either way when X approaches R, if X and R were calculated by different operations)

One address commands

14 CC K NN YYYYY (The rightmost bit of K is used with NN, the left-most two bits of K are used as tags for the Y address. Refer to detailed command explanations for details.

Index jumps with modification of index counter registers b (1 or 2)

(CC)

76 with b 1

77 with b 2

Add 2 to the contents of b (1 or 2). If N is greater than one half of b (1 or 2) jump to Y a nd leave b (1 or 2) advanced. Otherwise set b (1 or 2) equal to zero and continue with the next instruction. The effect is to go through the loop N + 1 times with the test at the end of the loop or N times with the test at the entrance to the loop.

Polynomial (CC)

75 Compute the polynomial $A_0 + A_1X + A_2X^2 + \ldots + A_nX^n$;

Where:

X, as a floating point number, must be prestored in register P.

Y is the address of the last constant, A_n, where the constants are floating point numbers stored each in two cells-mantissa, exponent-consecutively,

$A_0, A_1, \ldots, A_n$.

N is the degree of the equation.

The answer appears in R.

If N is less than one no operation will be performed.

Card instructions (CC)

57 Read N cards into cells consecutively starting at Y.

56 Punch N cards from cells consectutively starting at Y.

More detailed instructions and options available are given in the detailed instructions. The contents of register C cannot be punched by a 14 instruction.

In case of Bull failure set PAK equal, to

77430 for read, or

77007 for write.

ES will be restored followed by a 56 stop with V address that of the interrupted card instruction.

Sum

A sum is stored at 77777. Entrance at 77371 will calculate a new sum over the area 76000 to 77776 inclusive; place the new sum in the Q register; place the difference between the new sum and the stored sum in the accumulator, and come to a 56 00000 77377 stop. Restarting computer will replace the stored sum with the calculated sum and terminate with a 56 00000 30000 stop. In general when no changes have been made in the tape a zero difference in the accumulator indicates the routine is correctly stored on the Magnetic Drum.

PX 71900-10-(183)

II Detailed Specifications

This is an unpacked floating point package including card input and output, addition, subtraction, multiplication, division, two index registers and normalization; all with rounding. The floating point number representation is defined in the 1103 storage as follows;

$N = M \cdot 2^e$ where $1/2 \leq |m| < 1$ and $0 \leq |e| < 2^{35}$

M is scaled 35, e is scaled 0, where M is in the first of two consecutive cells and e in the second.

Examples:

1 = 200000000000 000000000001

or 377777777777 000000000000

-1 = 577777777777 000000000001

- 10 = 314631463146 777777777776

The card input and output will only handle a maximum decimal exponent of 99.

The number should be normalized at all times to gain the maximum accuracy because 35 bits are used in a normalized mantissa. However, the addition, subtraction, and multiplication operations will handle floating numbers not normalized and give normalized answers which are correct to a lesser number of places. The division operation requires a normalized divisor or will go to the alarm exit, where it will alarm print the divisor and halt with a 56 stop. A restart will then finish the current instruction using an incorrect answer, and No fault will be caused by a non-normalized dividend which will give a non-normalized quotient correct to a lesser number of places.

The number whose mantissa is in the accumulator scaled 35 and whose exponent is in 01775 will be normalized by the return jump 37 01741 01712.

The mantissa of the answer will be duplicated in 01774 and the accumulator, with the exponent in 01775, i.e. the normalized number will be left in R as for arithmetic operation answers.

To attempt greater convenience of programming subroutines and simpler explanation of operation; a core package, to operate from 01500 to 01777 inclusive, is independent of any subroutines used. The intention is that any subroutine may be easily assembled to use with it. It includes rounded operations for accuracy and the fundamental arithmetic operations. These last arithmetic operations can be used with interpretive instructions for normal programs or with return jump instructions for greater speed in loops and especially for construction of subroutines.

The package starts with a self contained arithmetic unit from 01633 to 01775 inclusive which will perform addition, subtraction, multiplication, division, and normalization operation on specified registers by return jump instructions. From 01500 to 01632 inclusive is an interpretive system which uses the smaller arithmetic package. From 77400 to 77751 on the drum is a card input, self contained and operated either by return jump instructions of by the interpretive system. From 76760 to 77357 is a similar card output. The package requires

00000 45 00000 (   )

00001 45 00000 01547

01500 - 01777 inclusive, package operation

76000 - 77777 inclusive for package and subroutine storage.

74000 - 75777 inclusive, ES image during some operations as for example card

PX 71900-10-(183)

input and output.

To attempt greater convenience of programming subroutines and simpler explanation of operation; a core package, to operate from 01500 to 01777 inclusive, is independent of any subroutines used. The intention is that any subroutine may be easily assembled to use with it. It includes rounded operations for accuracy and the fundamental arithmetic operations. These last arithmetic operations can be used with interpretive instructions for normal programs or with return jump instructions for greater speed in loops and especially for construction of subroutines.

The package starts with a self contained arithmetic unit from 01633 to 01775 inclusive which will perform addition, subtraction, multiplication, division, and normalization operating on specified registers by return jump instructions. From 01500 to 01632 inclusive is an interpretive system which uses the smaller arithmetic package. From 77400 to 77751 on the drum is a card input, self contained and operated either by return jump instructions of by the interpretive system. From 76760 to 77357 is a similar card output.

The package requires

00000 45 00000 (   )

00001 45 00000 01547

01500 - 01777 inclusive, package operation

76000 - 77777 inclusive for package and subroutine storage.

74000 - 75777 inclusive, ES image during some operations as for

example card input and output.

The tape reads into cells 76000 - 77777 inclusive 37 76000 76001 transfers from drum storage to E.S. cells 00000, 00001, and 01500 to 01757 inclusive and clears cells 01760 to 01777 for registers.

17 00000 77413 will prime read and punch cards initially and succeeding card movements will be automatic.

00002 - 01477 inclusive remain available for any program or added subroutines.

Registers

01764 P for storage of an
01765 operand for iteration
01766 B for storage of the
01767 (X) operand
01770 C for storage of the
01771 (Y) operand
01772 S for storage of an
01773 operand for iteratio
01774 R The answer for storage of result
01775
01776 b 1 Index Registers for
01777 b 2 address modificat

Secondary registers

01760 I contains the current or ~~ias xx xxxxxxtive~~ (14) instruction
01761 N is used as a counter, receives N from single address instructions
01762 Used as temporaries
01763 in normalization and arithmetic operations (except division)

PX 71900-10-(183)

Commands

Two address commands

14 CC XXXX YYYY

14 is the interpretive code

CC is the pseudo command code

XXXX (excluding the leftmost two bits) is the address of the operand placed in register B.

YYYY (excluding the leftmost two bits) is the address of the operand placed in register C.

The first (leftmost) bit of XXXX or YYYY is called a tag. If a one, it will cause a temporary addition of the contents of index register b 1 to that address during operation.

After the operation the instruction resumes its original unaltered form. The second (next) bit, if one, will add the contents of index register b 2 but otherwise operate as described for b 1. Both may be used in the same operation.

One address commands

14 CC KNN YYYYY

14 is the interpretive (1103) code.

CC is the pseudo command code

K is divided into two parts.

The leftmost 2 bits are used as tags on the Y address.

The first (leftmost) bit of K if a one will cause a temporary addition of the contents of index register b 1 to the Y address during operation. After the operation the instruction resumes its original unaltered form. The second (next) bit if a one will add the contents of index register b 2 but otherwise operate as above for b 1. Both tags may be used in the same operation.

The third (rightmost) bit of K will be used with NN as a number N. NN with the third bit of K is the number N placed in register N and usually used as a counter. Its maximum size is 177 octal or 127 decimal.

YYYYY is the address of the number placed in register C.

Commands and Registers

14 00 XXXX YYYY

Y + X to R

or 37 01741 01656

C + B to R

14 01 XXXX YYYY

Y - X to R

or 37 01741 01655

C - B to R

14 02 XXXX YYYY

Y . X to R

or 37 01741 01706

C · B to R

14 03 XXXX YYYY

Y/X to R

or 37 01741 01633

C/B to R

Registers

R final will be duplicated in C and the double extension of the R final mantissa will also be in the accumulator

P will be unchanged

For 14 type instructions

R initial will be in S and B

For 37 type instructions

S will be unchanged and duplicated in B.

Commands and Registers

14 04 XXXX YYYY

R initial + Y + X to R

14 05 XXXX YYYY

R initial + Y - X to R

14 06 XXXX YYYY

R initial + Y • X to R

or 37 01741 01617

S + Y • X to R

14 07 XXXX YYYY

R initial + Y/X to R

Registers

R final will be duplicated in C and the double extension of the R final mantissa will also be in the accumulator

P will be unchanged

For 14 type instructions

R initial will be in S and B

For 37 type instructions

S will be unchanged and duplicated in B

Commands and Registers

14 20 XXXX YYYY

Y + X to R, Y

or 37 01746 01656

C + B to R, see footnote*

14 21 XXXX YYYY

Y - X to R, Y

or 37 01746 01655

C - B to R, see footnote*

14 22 XXXX YYYY

Y · X to R, Y

or 37 01746 01706

C · B to R, see footnote*

14 23 XXXX YYYY

Y/X to R, Y

or 37 01746 01633

C/B to R, see footnote*

Registers

P will be unchanged

For 14 type instructions

R final will be duplicated in C and Y and the double extension of the R final mantissa will also be in the accumulator.

R initial will be in S and B

For 37 type instructions

* R final will be duplicated in C and also at the "Y" address taken from the last 14 instruction used and located in the V portion of cell 01744 (where it is subject to modification by the programmer)

PX 71900-10-(183)

The double extension of the final mantissa will also be in the accumulator.

S will be unchanged and duplicated in B.

Commands and Registers

14 24 XXXX YYYY

R + Y + X to R, Y

14 25 XXXX YYYY

R + Y - X to R, Y

14 26 XXXX YYYY

R + Y · X to R, Y

or 37 01746 01617

S + C · B to R, *

14 27 XXXX YYYY

R + Y/X to R, Y

Registers

P will be unchanged

For 14 type instructions

R final will be duplicated in C and Y and the double extension of the R final mantissa will also be in the accumulator.

R initial will be in S and B

For 37 type instructions

*R final will be duplicated in C and also at the "Y" address taken from the last 14 instruction used and located in the V portion of Cell 01744 (where it is subject to modification by the programmer) The double extension of the R final mantissa will also be in the accumulator

S will be unchanged and duplicated in B.

Commands and Registers

Threshold jump

14 17 XXXX YYYY

If X is greater than R jump to Y; otherwise continue with the next instruction (caution - due to rounding a jump may occur either way when X approaches R if X and R have been calculated by different operations)

R initial will be unchanged in R, S, and B

R - X will be in C and the mantissa of R - X will be in the accumulator

P will be unchanged.

Index jumps with modification of b index registers

14 76 KNN YYYYY with b 1

14 77 KNN YYYYY with b 2

Add 2 to the contents of b 1 or b 2. If N (last bit of K with NN from instruction is greater than b1 or b2 jump to y and leave b1 or b2 advanced.

Otherwise set b1 or b2 equal to zero and continue with the next instruction.

The contents of Y (2 cells) will be placed in register C.

N (from the last bit of K and NN) will be placed in register N and remain unchanged

R initial will be duplicated in S

R, P, and B will be unchanged.

Polynomial

14 75 KNN YYYYY

Compute the polynomial $A_0 + A_1X + A_2X^2 + \cdots + A_nX^n$; Where:

X must be prestored in register P

Y is the address of the last constant, A_N, where the constants are floating point numbers stored each in two cells - mantissa, exponent - consecutively, A_0, A_1, + ··· + A_N.

N is the degree of the equation.

The answer appears in R.

If N is less than one, no operation will be performed.

R and C will contain the answer.

S and B will contain A_0

P will contain X

N will be zero

The double extension of the mantissa of the answer will also be in the accumulator.

Commands and Registers

Card Instructions

14 57 KNN YYYYY

Read N cards (N is NN with the rightmost bit of K) into cells consecutively starting at Y in normalized floating point form.

14 56 KNN YYYYY

Punch N cards (N is NN with the rightmost bit of K) from cells consecutively starting at Y where numbers are stored in normalized floating point form.

Registers

Register N will contain the number N unchanged.

Register C will contain the first word stored at Y. In the case of read or punch this will be the contents of Y before read or punch - R, S, P, and B will be unchanged. Register C cannot be punched by a 14 instruction since Y is placed in C at the start of all 14 interpretations.

Return jump card options

Card read

37 77400 77401

AB UUUUU VVVVV

A. If a is equal to 4,5,6 or 7 do not print sum, otherwise print a sum which is characteristic of those cards.

B. If B is equal to 4,5,6 or 7 read in mantissas as integers to the zero scale factor, and store them in single consecutive cells. Ignore all exponents. Otherwise read in and store normal flosting point numbers in normalized form.

UUUUU Address for storage of the first data word.

PX 71900-10-(183)

VVVVV Number of floating point numbers to be read or with option number of mantissas stored as integers.

Card output

37 76760 76761

AB UUUUU VVVVV

A. If A is equal to 4,5,6 or 7 the cards will not be numbered, otherwise they will be numbered starting at one. Cell 77303 contains this counter and may be set at one less than the next positive card number desired.

B Is not used.

UUUUU Is the storage address of the first mantissa

VVVVV Is the number of floating point numbers to be punched.

Coding Notes

In case of Bull failure set PAK equal to:

77430 for read, or

77007 for write.

ES will be restored followed by a 56 stop with the V address that of the interrupted card instruction.

If cards are not primed at start of program:

17 00000 77433 will advance one read card.

17 00000 77342 will advance one punch card

17 00000 77413 will advance one read card and one punch card

If the mantissa is zero after any arithmetic operation the exponent will be set equal to zero. During card read the exponent of a zero mantissa will be retained.

In punching or writing cards if an un-normalized mantissa or an exponent with absolute value above 513 octal in any series is to be punched, that number will be punched as negative zero. In this last case an alarm listing (nf) of the address of the first such number in any card instruction and its mantissa will be made on the typewriter but the program will continue. The maximum absolute value of exponents punched on cards is 98 or 99 depending on the mantissa value.

In reading cards a punch in column number one and row number twelve of any card will act as a flag and terminate the read instruction at the end of that card with a normal return to the program. If N Is exhausted first an exit will occur. If it is desired to read a large number of cards with only a flag termination without regard to N, change the instruction at address 77445 to 11 01006 01314

Card Form Columns:

6, 21, 36, 51, 66 decimal point punched in card

7 - 16, 22-31, 37-46, 52-61, 67-76, ten decimal digit mantissas.

17, 32, 47, 62 77 sign of the mantissas.

18 - 19, 33 - 34, 48 - 49, 63 - 64, 78 - 79, exponents (power of ten) range - 99 to 99

20, 35, 50, 65, 80 signs of exponent.

Codes and Exits Available For Subroutines

Two address references from E. S. Whiffletree

Code	U Address of	V Address of
10, 30*		01632
11, 31*	01632	
12, 32*		01631
13, 33*	01631	
14, 34*		01627
15, 35*	01627	
16, 36*		01626

Note that the 10 to 17 series will set to prevent storage of the result in Y.

One address references from E. S. Whiffletree

Code	U address of	V Address of
60, 61, 62, 63		01527
64, 65, 66, 67	01527	
70		01505
71	01505	
72		01504
73	01504	
74		01531

One address references from drum Whiffletree

Code	U address of	V address of
40		76122
41	76122	
42		76121
43	76121	

Code	U Address of	V Address of
44		76117
45	76117	
46		76116
47	76116	
50		76113
51	76113	
52		76112
53	76112	
54		76110
55	76110	

The drum whiffletree stores E. S., transfers itself to E. S., and jumps to the chosen address.

PX 71900-10-(183)

SPUR ACCURACY TESTS:

20,000 Operations

07,21:

	Octal
Number in	246033531747 / 777777777774
Number out	246033536705 / 777777777774
	Decimal
Number in	.8108108108 01-
Number out	.8108109027 01-

04,01:

	Octal
Number in	400000000012 / 000000000005
Number out	400000077460 / 000000000005
	Decimal
Number in	.3199999998 - 02
Number out	.3199996967 - 02

03,22

	Octal
Number in	207002533507 / 000000000000
Number out	207002543105 / 000000000000
	Decimal
Number in	.5273641890 00
Number out	.5273643007 00

8th degree palynomial:	Exact	Computed
	1330157	1330156.999
	231285	231284.9999
	24835	24834.99999

SPUR time tests;

	Command	Time (Millisecounds)
14/	00	3.84
	04	5.69
	20	3.95
	24	5.73
	01	3.80
	05	5.74
	21	3.90
	25	5.80
	02	3.49
	06	5.33
	22	3.48
	26	5.28
	03	3.65
	07	5.45
	23	3.71
	27	5.60
	17	3.81
	76	1.47
	77	1.47
	75	27.0 (For 8th degree poly.)

37 01746 01706 (Comparable to 1422) 1.73 m.s.

37 01741 01633 (Comparable to 1403) 1.74 m.s.

PX 71900-10-(183)

CONVAIR

A DIVISION OF GENERAL DYNAMICS CORPORATION

SAN DIEGO

REPORT ZM 527-II

DATE 9-6-56

MODEL All

TITLE

S P U R

Single Precision Unpacked Rounded
Floating Point Package for

ERA 1103 Computer

Part II Codes

PREPARED BY L. W. Barton

GROUP Digital Computing Lab.

REFERENCE

CHECKED BY C. J. Swift, D. Parker
D. Bielsker

APPROVED BY

NO. OF PAGES 27

NO. OF DIAGRAMS

REVISIONS

NO.	DATE	BY	CHANGE	PAGES AFFECTED

FORM 1812 A-4

SPUR ACTIVATION

```
76000   01760   45 00000 30000   EXIT
76001   01761   11 76012 00000   SET F 0
76002   01762   11 76013 00001   SET F 1
76003   01763   75 30020 01765      THIS TO
76004   01764   11 76000 01760          ES
76005   01765   75 30260 01767      SPUR TO
76006   01766   11 76500 01500          ES
76007   01767   16 01760 01770   SET EXIT
76010   01770   75 10020 30000      CLEAR
76011   01771   11 01750 01760         EXIT
76012   01772   45 00000 00700   F 0
76013   01773   45 00000 01547   F 1
```

PX (1900-10-(183)

SPUR DRUM WHIFFLE TREE

```
76100          75 31777 76102      STORE
76101          11 00001 74001         ES
76102          75 30045 00004      THIS TO
76103          11 76104 00004         ES
76104   00004  44 00005 00014   WHIFFLETREE
76105   00005  44 00006 00011
76106   00006  44 00007 00010
76107   00007  44 00023 00036   CARD   READ WRITE
76110   00010  44 30000 30000   55 54
76111   00011  44 00012 00013
76112   00012  44 30000 30000   53 52
76113   00013  44 30000 30000   55 50
76114   00014  44 00015 00020
76115   00015  44 00016 00017
76116   00016  44 30000 30000   47 46
76117   00017  44 30000 30000   45 44
76120   00020  44 00021 00022
76121   00021  44 30000 30000   43 42
76122   00022  44 30000 30000   41 40
76123   00023  75 30352 00025      CARD READ
76124   00024  11 77400 00742         TO ES
76125   00025  16 01746 00770   SET EXIT
76126   00026  16 01746 00742      SET
76127   00027  31 00742 00000         FAULT
76130   00030  36 01264 77432            REPEAT
76131   00031  31 01761 00002      ASSEMBLE
76132   00032  35 01761 00742         PARAMETER
76133   00033  15 01601 00742            WORD
76134   00034  16 00032 00763   SET ACQUISITION
76135   00035  45 00000 00762
76136   00036  75 30400 00040      CARD WRITE
76137   00037  11 76760 00745         TO ES
```

```
76140   00040   16 01746 00772   SET EXIT
76141   00041   16 01746 00745       SET
76142   00042   31 00745 00000          FAULT
76143   00043   36 01310 77011             REPEAT
76144   00044   31 01761 00002       ASSEMBLE
76145   00045   35 01761 00745          PARAMETER
76146   00046   15 01601 00745             WORD
76147   00047   16 00045 00764   SET ACQUISITION
76150   00050   45 00000 00764
```

PX 71900-10-(183)

SPUR ES PACKAGE

76500	01500	44	30064	30060	
76501	01501	44	01516	01530	
76502	01502	44	01501	01503	
76503	01503	44	01504	01505	
76504	01504	44	30073	30072	
76505	01505	44	30071	30070	
76506	01506	44	01502	01500	
76507	01507	11	01750	01770	ONE ADDRESS INST.
76510	01510	16	01760	01770	ADDRESS
76511	01511	55	10000	00016	
76512	01512	51	01756	01761	NUMBER N
76513	01513	55	01760	10014	
76514	01514	37	01603	01570	SET ADDRESS TAGS
76515	01515	44	01506	76100	ES OR MD
76516	01516	15	01527	01524	SET
76517	01517	16	01564	01526	b 1
76520	01520	44	01521	01523	
76521	01521	15	01541	01524	SET
76522	01522	16	01566	01526	b 2
76523	01523	55	01761	10001	2 N
76524	01524	21	30000	01752	$b_i + 2$
76525	01525	42	10000	01747	JUMP TO Y
76526	01526	11	01750	30000	CLEAR
76527	01527	45	01776	01746	JUMP TO NI
76530	01530	15	01601	01537	SET Y ADDRESS
76531	01531	44	01533	30074	
76532	01532	37	01742	01617	ACCUMULATE MULT.
76533	01533	11	01764	01766	X FOR
76534	01534	11	01765	01767	MULT.
76535	01535	23	01537	01753	STEP
76536	01536	75	30002	01540	CONSTANT
76537	01537	11	30000	01772	A_i FOR ADD

76540	01540	41	01761	01532	TEST DEGREE
76541	01541	45	01777	01745	
76542	01542	11	01774	01770	
76543	01543	11	01775	01771	
76544	01544	37	01741	01655	R SUB X TO R
76545	01545	75	30002	01616	REPLACE
76546	01546	11	01772	01774	R
76547	01547	16	00000	01746	INTERPRET ENTRANCE
76550	01550	31	00000	00000	
76551	01551	34	01751	00017	
76552	01552	15	20000	01553	
76553	01553	11	30000	01760	STORE INST.
76554	01554	55	01760	10006	
76555	01555	44	01507	01556	ONE OR TWO ADDRESS
76556	01556	11	01760	10000	
76557	01557	51	01757	01770	Y ADDRESS
76560	01560	55	10000	00030	
76561	01561	51	01757	01766	X ADDRESS
76562	01562	55	01760	10014	
76563	01563	44	01564	01565	
76564	01564	21	01766	01776	X B1
76565	01565	44	01566	01567	
76566	01566	21	01766	01777	X B2
76567	01567	55	10000	00012	
76570	01570	44	01571	01572	ENT FROM ONE AD
76571	01571	21	01770	01776	Y B1
76572	01572	44	01573	01574	
76573	01573	21	01770	01777	Y B2
76574	01574	16	01770	01747	SET
76575	01575	16	01770	01744	Y JUMP
76576	01576	31	01770	00017	AND
76577	01577	15	20000	01601	TRANSFERS

PX 71900-10-(183)

```
76600   01600   75 30002 01602      Y TO
76601   01601   11 30000 01770         OP C
76602   01602   55 01760 10007
76603   01603   37 01603 01604   EXIT FOR ONE AD
76604   01604   11 01774 01772      R TO
76605   01605   11 01775 01773         S
76606   01606   44 01610 01607
76607   01607   21 01742 01752   NO TRANSMIT
76610   01610   31 01766 00017
76611   01611   15 20000 01613
76612   01612   75 30002 01614      X TO
76613   01613   11 30000 01766         OP B
76614   01614   44 01624 01615
76615   01615   44 01620 01621
76616   01616   46 01747 01746   Y OR N1 JUMP
76617   01617   75 00001 01706   JUMP TO MULT
76620   01620   16 01623 01741   SET FOR ACCUMULATE
76621   01621   44 01622 01623
76622   01622   44 01633 01706   DIVIDE  MULT.
76623   01623   44 01655 01656   SUB., ADD.
76624   01624   44 01625 01630
76625   01625   44 01626 01627
76626   01626   44 01542 30016   TJ, 16
76627   01627   44 30015 30014   15, 14
76630   01630   44 01631 01632
76631   01631   44 30013 30012   13, 12
76632   01632   44 30011 30010   11, 10
76633   01633   54 01766 20001      TEST FOR DIVISOR  DIVISION
76634   01634   43 20000 77752         ZERO OR UNNORMALIZED
76635   01635   11 01771 20000      EXPONENT
76636   01636   36 01767 01775         DIFFERENCE
76637   01637   12 01770 20000      COMPARE
```

```
76640   01640   12 01766 10000        MANTISSA
76641   01641   36 10000 20000           SIZE
76642   01642   46 01643 01645
76643   01643   54 01770 20043     SHIFT
76644   01644   45 00000 01647        OR ADJUST
76645   01645   21 01775 01751           EXPONENT
76646   01646   54 01770 20042              AND SHIFT
76647   01647   73 01766 01774  DIVIDE
76650   01650   32 01750 00001     TEST
76651   01651   12 01767 10000        FOR
76652   01652   42 10000 01734           ROUND
76653   01653   21 01774 01751  ROUND
76654   01654   45 00000 01730
76655   01655   13 01766 01766  SUBTRACTION
76656   01656   11 01767 01775  ADDITION
76657   01657   23 10000 10000  CLEAR A, Q
76660   01660   16 10000 01700  ERASE SHIFT
76661   01661   43 01770 01700  Y ZERO TEST
76662   01662   43 01766 01676  X ZERO TEST
76663   01663   23 01767 01771  EXPONENT DIFFERENCE
76664   01664   12 20000 20000
76665   01665   42 01755 01670  36 COMPARISON
76666   01666   41 01767 01700  Y OR X
76667   01667   45 00000 01676
76670   01670   16 20000 01700  SET SHIFT
76671   01671   41 01767 01674  Y OR X
76672   01672   11 01766 10000  X TO Q
76673   01673   45 00000 01677
76674   01674   11 01770 10000  Y TO Q
76675   01675   75 00001 01700
76676   01676   11 01771 01775  Y EXP
76677   01677   11 01770 01766  Y TO X
```

```
76700   01700   54 01766 30000   SHIFT X
76701   01701   35 10000 01774
76702   01702   45 00000 01713   TO NORMALIZE
76703   01703   23 01775 01755      A LEFT
76704   01704   54 01774 00044         NOT SIGNIFICANT
76705   01705   47 01714 01734   ZERO TEST
76706   01706   11 01771 20000   MULTIPLICATION
76707   01707   35 01767 20000   ADD EXPONENTS
76710   01710   36 01754 01775
76711   01711   71 01770 01766   PRODUCT
76712   01712   11 20000 01774     TEST
76713   01713   43 20000 01703         EXTENTION
76714   01714   11 01750 01762   CLEAR FOR SF
76715   01715   11 01751 10000   POS. FLAG
76716   01716   46 01717 01720   SIGN
76717   01717   13 10000 10000   NEG. FLAG
76720   01720   11 10000 01763   FOR ROUNDING
76721   01721   74 20000 01762   SCALE FACTOR
76722   01722   11 20000 01774
76723   01723   46 01724 01725      ADJUST
76724   01724   13 10000 10000         FOR SIGN
76725   01725   21 01775 01762   ADJUST FOR SF
76726   01726   44 01727 01734
76727   01727   21 01774 01763   ROUND
76730   01730   43 20000 01734   OVERFLOW TEST
76731   01731   32 01750 00107      ADJUST
76732   01732   11 20000 01774         FOR
76733   01733   21 01775 01751            OVERFLOW
76734   01734   11 01774 20000      ZERO
76735   01735   47 01737 01736         TEST
76736   01736   11 20000 01775   ERASE EXPONENT
76737   01737   75 30004 01741   SET FOR
```

76740	01740	11	01772	01766	ITERATION
76741	01741	37	01741	01742	FOR ITERATION
76742	01742	37	01742	01743	FOR BYPASS
76743	01743	75	30002	01745	STORE
76744	01744	11	01774	01774	R TO Y
76745	01745	45	00000	01746	FOR TRACE
76746	01746	45	00000	30000	TO NEXT INST.
76747	01747	45	00000	30000	TO Y
76750	01750	00	00000	00000	CONSTANTS
76751	01751	00	00000	00001	
76752	01752	00	00000	00002	
76753	01753	00	00002	00000	
76754	01754	00	00000	00043	
76755	01755	00	00000	00044	
76756	01756	00	00000	00177	
76757	01757	00	00000	01777	CONSTANTS
	01760	00	00000	00000	I FOR INSTRUCTION
	01761	00	00000	00000	N COUNTER
	01762	00	00000	00000	TEMPORARY
	01763	00	00000	00000	TEMPORARY
	01764	00	00000	00000	P FOR OPERAND
	01765	00	00000	00000	STORAGE FOR ITERATION
	01766	00	00000	00000	B FO
	01767	00	00000	00000	X OPERAND
	01770	00	00000	00000	C FOR
	01771	00	00000	00000	Y OPERAND
	01772	00	00000	00000	S FOR OPERAND
	01773	00	00000	00000	STORAGE FOR ITERATION
	01774	00	00000	00000	R FOR
	01775	00	00000	00000	THE RESULT
	01776	00	00000	00000	1 INDEX
	01777	00	00000	00000	2 REGISTERS

PX 71900-10-(183)

SPUR CARD OUTPUT

```
76760   00745   56 00000 30000
76761   00746   75 31777 76763      STORE
76762   00747   11 00001 74001        E S
76763   00750   75 30400 00752      ROUTINE
76764   00751   11 76760 00745        TO E S
76765   00752   31 00745 00000      SET
76766   00753   36 01310 77011       REPEAT
76767   00754   16 00745 00772      SET
76770   00755   21 00772 01310       EXIT
76771   00756   55 00745 20025      MODIFY
76772   00757   44 00761 00760       E S
76773   00760   32 00777 00000        ADDRESSES
76774   00761   55 20000 00017         TO STORAGE
76775   00762   37 00762 00763
76776   00763   16 20000 00764      ACQUIRE
76777   00764   71 01310 30000       CONTROL WORD
77000   00765   55 20000 00006      EXAMINE
77001   00766   37 00762 00757       FOR E S
77002   00767   55 20000 00017        ADDRESSES
77003   00770   37 01001 01006      TO SUBROUTINE
77004   00771   11 01270 77303      STORE CARD NUMBER
77005   00772   75 31777 30000      RESTORE
77006   00773   11 74001 00001         E S
77007   00774   75 31777 77011      RESTORE
77010   00775   11 74001 00001       E S
77011   00776   30 00000 00000        REPEAT
77012   00777   74 00000 00000      E S STORAGE
77013   01000   30 00000 00000      REPEAT LAST INST
77014   01001   45 00000 30000      EXIT
77015   01002   16 01001 01005
77016   01003   21 01001 01310
77017   01004   36 01344 01000
```

77020	01005	71	01310	30000	
77021	01006	15	20000	01026	FIRST ADDRESS
77022	01007	13	20000	01347	FLAG
77023	01010	11	01310	01350	FLAG
77024	01011	31	20000	00025	EXTRACT
77025	01012	31	20000	00063	AND STORE
77026	01013	36	01310	01351	N-1
77027	01014	46	01001	01015	TEST FOR ZERO
77030	01015	17	00000	01327	PICK WRITE CARD
77031	01016	31	01351	00000	
77032	01017	11	20000	01352	N-1 PER CARD
77033	01020	11	01325	10000	WRITE
77034	01021	42	01342	01024	
77035	01022	11	01305	01352	5 PER CARD
77036	01023	11	01324	10000	WRITE AND PICK CARD
77037	01024	17	00000	10000	CARD INSTRUCTION
77040	01025	75	30012	01027	5 WORDS TO
77041	01026	11	30000	01353	STORAGE
77042	01027	75	10045	01031	CLEAR FOR
77043	01030	11	01307	01376	CARD IMAGE
77044	01031	15	01266	01050	SET ACQUISITION
77045	01032	11	01340	01374	BIT
77046	01033	11	01265	01375	BIT INSTRUCTION
77047	01034	41	01347	01047	IDENT NUMBER TEST
77050	01035	11	01332	01374	BIT
77051	01036	21	01270	01310	COUNT CARDS
77052	01037	42	01315	01041	SIZE OF WORD
77053	01040	31	01341	00000	SET 99999
77054	01041	32	01307	00043	INTEGER
77055	01042	32	01265	00000	TO
77056	01043	73	01315	01365	FRACTION
77057	01044	54	01365	00001	2 EXP 36

PX 71900-10-(183)

```
77060   01045   11 01305 01376      N-1 DIGITS
77061   01046   37 01200 01165      IDENT NUMBER
77062   01047   75 30002 01053      MANTISSA
77063   01050   11 30000 01365       AND EXPONENT
77064   01051   11 01307 01366      ZERO EXPONENT
77065   01052   45 00000 01156
77066   01053   11 01365 01370      MANTISSA SIGN FLAG
77067   01054   12 01366 20000      EXPONENT
77070   01055   42 01301 01067       FLOATING TEST
77071   01056   41 01350 01062      FIRST TIME
77072   01057   11 01264 01370      MANTISSA FLAG
77073   01060   75 10002 01156      SET EQUAL
77074   01061   11 01307 01365       TO ZERO
77075   01062   13 01322 01350      SET ALARM FLAG
77076   01063   11 01026 01345
77077   01064   15 01050 01065      ACQUIRE
77100   01065   11 30000 01346       MANTISSA
77101   01066   45 00000 01057
77102   01067   21 01365 01307
77103   01070   47 01071 01051      ZERO MANTISSA
77104   01071   46 01072 01073
77105   01072   13 01365 01365      NEGATIVE MANTISSA
77106   01073   16 01323 01123      SET SHIFT TO 33
77107   01074   32 01307 00001      EXP 36
77110   01075   43 20000 01056      FLOATING TEST
77111   01076   11 01307 01373      CLEAR FOR DECIMAL EXP
77112   01077   11 01366 20000      EXPONENT
77113   01100   46 01101 01117       SIGN
77114   01101   23 01373 01342      10 EXP 5 ADJUSTMENT
77115   01102   11 01307 01372      CLEAR FOR 74
77116   01103   71 01315 01365      X 10 EXP 5
77117   01104   74 20000 01372
```

77120	01105	11	20000	01365	
77121	01106	46	01107	01114	TEST FOR ROUND
77122	01107	21	01365	01310	ROUND
77123	01110	43	20000	01114	MANTISSA
77124	01111	31	01365	00107	AND
77125	01112	11	20000	01365	ADJUST
77126	01113	21	01372	01310	
77127	01114	21	01366	01372	ADJUST BSF
77130	01115	46	01101	01116	SIGN OF BSF
77131	01116	31	01366	00000	DIVIDE EXP
77132	01117	73	01323	01367	BY 33
77133	01120	11	20000	01366	EXPONENT REMAINDER
77134	01121	55	01365	00001	MANTISSA
77135	01122	45	00000	01152	
77136	01123	31	01365	30000	MANTISSA SHIFT
77137	01124	11	20000	01371	STORE AR
77140	01125	34	20000	00044	ERASE
77141	01126	11	20000	01365	STORE AL
77142	01127	75	20013	01426	TEST FOR LARGER
77143	01130	42	01310	01131	POWER OF TEN
77144	01131	51	01330	10000	
77145	01132	31	01311	00000	10 EXP 10
77146	01133	36	10000	10000	
77147	01134	35	01373	01373	INCREASE EXP
77150	01135	31	10000	00017	N X 2 EXP 15
77151	01136	35	01263	01137	
77152	01137	30	00000	00000	ACQUIRE DIVISOR
77153	01140	11	10000	01372	
77154	01141	31	01365	00044	RESTORE
77155	01142	32	01371	00000	A
77156	01143	73	10000	10000	2 EXP 35
77157	01144	55	10000	00001	2 EXP 36

PX 71900-10-(183)

```
77160   01145   31 20000 00001      DETERMINE
77161   01146   42 01372 01150       LAST
77162   01147   27 10000 01310        BIT
77163   01150   11 10000 01365      ANSWER X 2 EXP 36
77164   01151   37 01151 01152
77165   01152   41 01367 01123      HIGHER ORDER DIGIT
77166   01153   16 01366 01123      LOWER ORDER SHIFT
77167   01154   37 01151 01123      LOWER ORDER DIGIT
77170   01155   11 01373 01366      DECIMAL EXPONENT
77171   01156   21 01050 01326      STEP
77172   01157   21 01026 01326      STEP
77173   01160   37 01200 01174      SHIFT FOR PERIOD
77174   01161   12 01366 20000      EXPONENT
77175   01162   11 01333 01376      TALLY
77176   01163   73 01311 01372      DIGITS
77177   01164   11 20000 01371       OF EXP
77200   01165   31 01365 00002      EXTRACT
77201   01166   32 01365 00001       AND
77202   01167   11 20000 01365        POSITION
77203   01170   34 20000 00063         DIGIT
77204   01171   35 01375 01172      ASSEMBLE INST
77205   01172   30 00000 00000      SET BIT
77206   01173   37 01173 01174
77207   01174   55 01374 00043      SHIFT BIT
77210   01175   44 01176 01177
77211   01176   21 01375 01334      ADVANCE FIELD
77212   01177   41 01376 01165      TEST FOR END
77213   01200   37 01200 01201
77214   01201   11 01370 10000
77215   01202   44 01203 01205
77216   01203   13 01331 20000
77217   01204   37 01173 01171      SIGN
```

```
77220  01205  55 01374 00043    SHIFT BIT
77221  01206  37 01206 01207
77222  01207  31 01372 00017    TENS DIGIT
77223  01210  37 01200 01171
77224  01211  31 01371 00017    UNITS DIGIT
77225  01212  37 01200 01171
77226  01213  11 01366 01370
77227  01214  37 01206 01201
77230  01215  37 01215 01216
77231  01216  41 01352 01047    WORDS PER CARD
77232  01217  11 01335 01371    SIT FOR 12 ROWS
77233  01220  21 01377 01336
77234  01221  21 01411 01336
77235  01222  21 01404 01336
77236  01223  21 01413 01337
77237  01224  21 01425 01337
77240  01225  21 01420 01337
77241  01226  16 01267 01233
77242  01227  16 01127 01234
77243  01230  15 01267 01231
77244  01231  55 30000 00010
77245  01232  77 00000 10000
77246  01233  77 10000 30000
77247  01234  77 10000 30000
77250  01235  23 01231 01331
77251  01236  23 01233 01310
77252  01237  23 01234 01310
77253  01240  41 01371 01231    TEST FOR END OF CARD
77254  01241  23 01351 01342    TEST FOR
77255  01242  46 01243 01016     END
77256  01243  41 01350 01001    FLOATING FLAG
77257  01244  75 10004 01246
```

77260	01245	61	00000	01302	
77261	01246	11	01305	01371	
77262	01247	55	01345	10011	
77263	01250	31	01264	00000	
77264	01251	52	01343	01252	
77265	01252	30	00000	00000	
77266	01253	55	10000	00003	
77267	01254	41	01371	01250	
77270	01255	37	01255	01256	
77271	01256	55	01346	10003	
77272	01257	61	00000	01305	
77273	01260	11	01335	01371	
77274	01261	37	01255	01250	
77275	01262	45	00000	01001	
77276	01263	55	01310	10001	
77277	01264	61	00000	01271	
77300	01265	21	01401	01374	
77301	01266	00	01353	00000	
77302	01267	00	01442	01412	
77303	01270	00	00000	00000	CARD COUNTER
77304	01271	00	00000	00037	
77305	01272	00	00000	00052	
77306	01273	00	00000	00074	
77307	01274	00	00000	00070	
77310	01275	00	00000	00064	
77311	01276	00	00000	00062	
77312	01277	00	00000	00066	
77313	01300	00	00000	00072	
77314	01301	00	00000	00513	
77315	01302	00	00000	00045	
77316	01303	00	00000	00006	
77317	01304	00	00000	00026	

77320	01305	00	00000	00004	
77321	01306	00	00000	00143	EXCESS
77322	01307	00	00000	00000	
77323	01310	00	00000	00001	
77324	01311	00	00000	00012	
77325	01312	00	00000	00144	
77326	01313	00	00000	01750	
77327	01314	00	00000	23420	
77330	01315	00	00003	03240	
77331	01316	00	00036	41100	
77332	01317	00	00461	13200	
77333	01320	00	05753	60400	
77334	01321	00	73465	45000	
77335	01322	11	24027	62000	
77336	01323	00	00000	00041	
77337	01324	00	00000	00112	
77340	01325	00	00000	00102	
77341	01326	00	00002	00000	
77342	01F27	00	00000	00110	
77343	01330	00	00000	00077	
77344	01331	00	00001	00000	
77345	01332	40	00000	00000	
77346	01333	00	00000	00011	
77347	01334	00	00014	00000	
77350	01335	00	00000	00013	
77351	01336	01	00001	00001	
77352	01337	00	00100	00100	
77353	01340	01	00000	00000	
77354	01341	00	00003	03237	
77355	01342	00	00000	00005	
77356	01343	00	00000	00007	
77357	01344	00	00000	00002	

PX 71900-10-(183)

SPUR SUM AND MT INSTRUCTION

```
70000          45 00000 70006    SROK FROM MT 3
70001          45 00000 70100    FLIP FROM MT 3
70002          45 00000 71404    SPUR FROM MT 3
70003          45 00000 77371    SUM SPUR
70004          45 00000 77376    CHANGE SUPUR SUM
70005          45 00000 70071    ERA MAINT. FROM MT 3
77371          31 70000 00000    START SUM
77372          75 23777 77374
77373          32 70001 00000
77374          75 21777 71400
77375          32 76000 00000
77376          75 00001 77371    CHANGE
77377          11 10000 77777     SUM
71400          34 77303 00000    REMOVE
71401          34 77400 00000     VARIABLE
71402          75 20013 71426      FROM
71403          34 76110 00000       SUM
71404          11 71423 00000    SET F
71405          75 31777 71407    STORE
71406          11 00001 74001     ES
71407          75 30015 00411    THIS
71410          11 71411 00411     TO ES
71411   00411  64 30001 00000    READ ONE BLOCK
71412   00412  11 00000 20000    TEST FOR
71413   00413  43 00425 00421     ZERO BLOCK
71414   00414  31 71424 00052    PRINT
71415   00415  61 00000 20000     TAPE
71416   00416  34 20000 00006      ALARM
71417   00417  47 00415 00420
71420   00420  56 00000 00411    TRY AGAIN
71421   00421  66 30400 00000    ADVANCE TO SPUR
71422   00422  64 30001 00000    READ IN FIRST BLOCK
```

77320	01305	00	00000	00004	
77321	01306	00	00000	00143	EXCESS
77322	01307	00	00000	00000	
77323	01310	00	00000	00001	
77324	01311	00	00000	00012	
77325	01312	00	00000	00144	
77326	01313	00	00000	01750	
77327	01314	00	00000	23420	
77330	01315	00	00003	03240	
77331	01316	00	00036	41100	
77332	01317	00	00461	13200	
77333	01320	00	05753	60400	
77334	01321	00	73465	45000	
77335	01322	11	24027	62000	
77336	01323	00	00000	00041	
77337	01324	00	00000	00112	
77340	01325	00	00000	00102	
77341	01326	00	00002	00000	
77342	01F27	00	00000	00110	
77343	01330	00	00000	00077	
77344	01331	00	00001	00000	
77345	01332	40	00000	00000	
77346	01333	00	00000	00011	
77347	01334	00	00014	00000	
77350	01335	00	00000	00013	
77351	01336	01	00001	00001	
77352	01337	00	00100	00100	
77353	01340	01	00000	00000	
77354	01341	00	00003	03237	
77355	01342	00	00000	00005	
77356	01343	00	00000	00007	
77357	01344	00	00000	00002	

PX 71900-10-(183)

SPUR SUM AND MT INSTRUCTION

70000		45	00000	70006	SROK FROM MT 3
70001		45	00000	70100	FLIP FROM MT 3
70002		45	00000	71404	SPUR FROM MT 3
70003		45	00000	77371	SUM SPUR
70004		45	00000	77376	CHANGE SUPUR SUM
70005		45	00000	70071	ERA MAINT. FROM MT 3
77371		31	70000	00000	START SUM
77372		75	23777	77374	
77373		32	70001	00000	
77374		75	21777	71400	
77375		32	76000	00000	
77376		75	00001	77371	CHANGE
77377		11	10000	77777	SUM
71400		34	77303	00000	REMOVE
71401		34	77400	00000	VARIABLE
71402		75	20013	71426	FROM
71403		34	76110	00000	SUM
71404		11	71423	00000	SET F
71405		75	31777	71407	STORE
71406		11	00001	74001	ES
71407		75	30015	00411	THIS
71410		11	71411	00411	TO ES
71411	00411	64	30001	00000	READ ONE BLOCK
71412	00412	11	00000	20000	TEST FOR
71413	00413	43	00425	00421	ZERO BLOCK
71414	00414	31	71424	00052	PRINT
71415	00415	61	00000	20000	TAPE
71416	00416	34	20000	00006	ALARM
71417	00417	47	00415	00420	
71420	00420	56	00000	00411	TRY AGAIN
71421	00421	66	30400	00000	ADVANCE TO SPUR
71422	00422	64	30001	00000	READ IN FIRST BLOCK

71423	00423	45	00000	00002	JUMP TO FIRST BLOCK
71424	00424	45	01301	52056	FLEX TAPE
71425	00425	45	07777	00002	LOCATOR ON ZERO BLOCK
71426		34	77432	00000	FURTHER
71427		34	76760	00000	SUM
71430		34	77011	00000	ADJUSTMENTS
71431		11	20000	20000	
71432		11	20000	10000	SUM TO Q
71433		36	77777	20000	SUBTRACT STORED
71434		47	71435	71445	SUM, TEST
71435		31	71466	00052	SPUR FOR PRINT
71436		61	00000	20000	PRINT
71437		34	20000	00006	WORD
71440		47	71436	71441	
71441		37	71441	71442	SWITCH
71442		31	71470	00052	PRINT
71443		37	71441	71436	NO SUM
71444		56	10000	71404	TO READ FROM MT
71445		37	71441	71435	PRINT SPUR
71446		31	71467	00052	PRINT
71447		37	71441	71436	SUM OK
71450		56	00000	71451	TO CLEAR
71451		75	17000	71453	START CLEAR
71452		11	20000	40000	
71453		75	17000	71455	
71454		11	20000	47000	
71455		75	17000	71457	
71456		11	20000	56000	
71457		75	13000	71472	
71460		11	20000	65000	
71461		75	11777	71463	
71462		11	20000	00001	

71463	31	71471	00052	PRINT
71464	37	71441	71436	CLEAR
71465	57	00000	00000	
71466	45	24153	41204	FLEX SPUR
71467	24	34070	40336	FLEX SUM OK
71470	06	03042	43407	FLEX NO SUM
71471	04	16112	03012	FLEX CLEAR
71472	75	12000	71461	CLEAR 74000
71473	11	20000	74000	75777

SPUR CARD INPUT

77400	00742	56	00000	30000	EXIT
77401	00743	75	31777	77403	STORE
77402	00744	11	00001	74001	ES
77403	00745	75	30352	00747	ROUTINE
77404	00746	11	77400	00742	TO ES
77405	00747	11	00742	20000	SET
77406	00750	36	01264	77432	REPEAT
77407	00751	35	01305	20000	SET
77410	00752	16	20000	00770	EXIT
77411	00753	55	00742	20025	TEST FOR
77412	00754	44	00757	00756	ES
77413	00755	00	00000	00114	ADDRESSES
77414	00756	32	00776	00000	ADDRESS MODIFICATION
77415	00757	55	20000	00017	
77416	00760	37	00760	00761	
77417	00761	16	20000	00763	SET ACQUISITION
77420	00762	11	00777	01024	ERASE INSTRUCTION
77421	00763	71	01264	30000	ACQUIRE CONTROL WORD
77422	00764	55	20000	00006	
77423	00765	37	00760	00754	X
77424	00766	55	20000	00017	
77425	00767	37	01001	01004	TP SIBRPITOME
77426	00770	75	31777	30000	RESTORE E S
77427	00771	11	74001	00001	AND EXIT
77430	00772	75	31777	77432	RESTORE E S
77431	00773	11	74001	00001	AND
77432	00774	30	00000	00000	REPEAT
77433	00775	02	00000	00104	
77434	00776	74	00000	00000	
77435	00777	11	00000	00000	
77436	01000	30	00000	00000	REPEAT
77437	01001	45	00000	30000	EXIT

PX 71900-10-(183)

77440	01002	16	01001	01003	
77441	01003	71	01264	30000	ACQUIRE CONTROL WORD
77442	01004	13	20000	01352	PRINT SUM FLAG
77443	01005	11	01263	01314	CLEAR
77444	01006	11	01263	01315	CELLS
77445	01007	16	20000	01314	NUMBER OF WORDS N
77446	01010	55	20000	00003	
77447	01011	13	20000	01351	READ AS INTEGER FLAG
77450	01012	31	01255	00000	SET FINAL
77451	01013	35	01277	01227	TRANSFER
77452	01014	11	01305	01356	SET TRANSFER
77453	01015	11	01265	01357	STEPS
77454	01016	44	01017	01022	
77455	01017	23	01227	01313	MODIFICATINNS
77456	01020	11	01264	01356	FOR INTEGER
77457	01021	23	01357	01261	STORAGE
77460	01022	55	10000	00021	SET DATA
77461	01023	16	10000	01230	STORAGE
77462	01024	21	01001	01264	SET EXIT
77463	01025	36	01305	01000	SET REPEAT
77464	01026	23	01314	01264	N-1
77465	01027	46	01001	01030	EXIT IF N 0
77466	01030	41	01352	01032	SUM TEST
77467	01031	61	00000	01303	CARRIAGE RETURN
77470	01032	17	00000	01300	READ AND PICK CARC
77471	01033	11	01260	01350	LINE DIGIT 9
77472	01034	16	01256	01224	SET TEMPORARY STORAGE
77473	01035	75	10011	01037	CLEAR
77474	01036	11	01263	01316	MATRIX
77475	01037	76	00000	01361	
77476	01040	76	10000	10000	
77477	01041	76	10000	01360	

77500	01042	37	01042	01043	
77501	01043	54	01361	00034	
77502	01044	11	01257	01054	SET 1ST STORAGE
77503	01045	11	01304	01327	SET INDEX 3
77504	01046	31	01306	00024	2 EXP 35
77505	01047	32	01263	00004	SHIFT 4
77506	01050	44	01051	01052	TEST BIT 0
77507	01051	32	01350	00000	ADD LINE DIGIT
77510	01052	46	01053	01047	TEST DIGITS 9
77511	01053	31	20000	00000	CLEAR A LEFT
77512	01054	30	00000	00000	STORE MATRIX WORD
77513	01055	21	01054	01307	STEP STORAGE
77514	01056	41	01327	01046	4 TIMES
77515	01057	37	01057	01060	
77516	01060	11	01360	10000	
77517	01061	37	01057	01045	
77520	01062	11	01361	10000	
77521	01063	37	01057	01046	
77522	01064	37	01064	01065	
77523	01065	23	01350	01264	REDUCE LINE DIGIT
77524	01066	46	01067	01037	TEST FOR 11 ROW
77525	01067	11	01264	01350	
77526	01070	37	01064	01037	11 ROW
77527	01071	37	01042	01037	12 ROW, DUMMY
77530	01072	44	01073	01075	TEST EXIT FLAG
77531	01073	11	01302	01314	OVERRIDE INDEX
77532	01074	11	01255	01350	SET FLAG
77533	01075	31	01314	00000	N-1 REMAINDER
77534	01076	42	01261	01100	TEST LAST CARD
77535	01077	17	00000	01300	READ AND PICK CARD
77536	01100	11	01302	01332	SET INDEX FOR WORD CHANGE
77537	01101	15	01257	01123	SET FOR 1ST EXTRACTION

PX 71900-10-(183)

```
77540   01102   55 01316 00024      SHIFT FOR IDENT NUMBER
77541   01103   11 01302 01347      SET FOR 5 DATA WORDS
77542   01104   31 01304 00000      TEST FOR
77543   01105   42 01314 01115        LESS THAN
77544   01106   11 01314 01347        5 WORDS
77545   01107   31 01314 00017      N-1
77546   01110   35 01306 10000      TEST FOR
77547   01111   41 01351 01113        INTEGER
77550   01112   55 10000 00001          OPTION
77551   01113   31 01255 00000      SET SMALLER
77552   01114   35 10000 01227        STORAGE
77553   01115   37 01124 01120      SPACE FOR PERIOD
77554   01116   11 01260 01331      SET TALLY   9
77555   01117   11 01263 01327      CLEAR
77556   01120   41 01332 01123      TEST TO
77557   01121   21 01123 01306       CHANGE MATRIX
77560   01122   11 01262 01332        WORD
77561   01123   55 30000 00004      PSSITION DIGIT
77562   01124   37 01124 01125
77563   01125   31 01327 00002      N X 10
77564   01126   32 01327 00001       ADD
77565   01127   52 01301 01327        DIGIT
77566   01130   41 01331 01120      TEST N COMPLETE
77567   01131   11 01327 01334      SET EXPONENT FOR N   3
77570   01132   37 01124 01120       ACQUIRE
77571   01133   51 01301 20000       SZ¤PN
77572   01134   11 01265 01330        FLAG
77573   01135   47 01136 01140      TEST SIGN
77574   01136   13 01327 01334      SET EXPONENT FOR N   0
77575   01137   13 01265 01330      -FLAG
77576   01140   37 01140 01141
77577   01141   11 01327 01354      MANTISSA X 10 EXP 10
```

77600	01142	11	01330	01355	MANTISSA FLAG
77601	01143	11	01264	01331	SET TALLY
77602	01144	37	01140	01117	EXPONENT INFORMATION
77603	01145	41	01351	01217	TEST INTEGER OPTION
77604	01146	11	01263	01333	CLEAR
77605	01147	11	01354	20000	TEST N
77606	01150	47	01151	01223	FOR 0
77607	01151	23	01327	01330	EXPONENT
77610	01152	12	20000	20000	ADJUSTMENT.
77611	01153	73	01265	01353	TENS DIGIT
77612	01154	31	20000	00017	UNITS DIGIT
77613	01155	35	01254	01215	UNITS POWER
77614	01156	11	20000	01251	OF 10
77615	01157	23	01334	01265	ADJUST EXPONENT
77616	01160	11	01276	01327	10 EXP 10
77617	01161	11	01310	01334	ADJUST AXPONENT
77620	01162	46	01214	01250	SIGN OF EXPONENT
77621	01163	23	01334	01311	ADJUST EXPONENT
77622	01164	31	01354	00000	N
77623	01165	73	01327	01333	N ADJUSTMENT
77624	01166	31	20000	00043	FOR NEGATIVE
77625	01167	73	01327	10000	EXPONENT
77626	01170	55	10000	00001	DETERMINE
77627	01171	32	01263	00001	LAST
77630	01172	42	01327	01174	BIT
77631	01173	27	10000	01264	OF N
77632	01174	31	10000	00044	ASSEMBLE
77633	01175	32	01333	00044	M
77634	01176	11	01263	01331	CLEAR
77635	01177	74	20000	01331	SCALE FACTOR N
77636	01200	11	20000	01354	STORE N
77637	01201	46	01202	01207	TEST FOR ROUNDING

PX 71900-10-(183)

```
77640   01202   21 01354 01264      ROUND
77641   01203   43 20000 01207      ADJUST FOR
77642   01204   32 01263 00107       OVERFLOW
77643   01205   11 20000 01354        IF
77644   01206   21 01334 01264         NECESSARY
77645   01207   31 01331 00000      ADJUST
77646   01210   42 01303 01212
77647   01211   23 01334 01312       SCALE
77650   01212   21 01334 01331        FACTOR
77651   01213   37 01213 01214
77652   01214   41 01353 01163      TEST TENS DIGIT
77653   01215   30 00000 00000      POWER FOR UNITS DIGIT
77654   01216   37 01213 01163      ADJUST FOR UNITS DIGIT
77655   01217   11 01354 01333      N
77656   01220   41 01355 01222      SIGN OF N
77657   01221   13 01333 01333      NEGATIVE
77660   01222   21 01315 01333      SUM
77661   01223   75 30002 01225      STORE
77662   01224   11 01333 30000       TEMPORARY
77663   01225   21 01224 01356      STEP STORAGE
77664   01226   41 01347 01115      TEST END OF CARD
77665   01227   30 00000 00000      FINAL
77666   01230   11 01335 30000       STORAGE
77667   01231   21 01230 01357      STEP
77670   01232   23 01314 01261      TEST FOR
77671   01233   46 01234 01033       END
77672   01234   41 01352 01001      TEST PRINT SUM
77673   01235   31 01263 00000      CLEAR
77674   01236   75 00006 01240      SUM
77675   01237   32 01315 00014
77676   01240   31 20000 00030
77677   01241   32 01264 00004
```

77700	01242	61	00000	20000	
77701	01243	34	20000	00000	
77702	01244	47	01241	01001	PRINTING
77703	01245	71	01354	01327	ADJUSTMENT
77704	01246	37	01213	01176	OF N
77705	01247	37	01247	01250	AND
77706	01250	41	01353	01245	EXPONENT
77707	01251	30	00000	00000	FOR
77710	01252	37	01247	01245	POSITIVE
77711	01253	45	00000	01217	SCALE FACTOR
77712	01254	11	01264	01327	CONSTANTS
77713	01255	75	30000	01231	
77714	01256	00	00000	01335	
77715	01257	35	01316	01316	
77716	01260	00	00000	00011	
77717	01261	00	00000	00005	
77720	01262	00	00000	00010	
77721	01263	00	00000	00000	
77722	01264	00	00000	00001	
77723	01265	00	00000	00012	
77724	01266	00	00000	00144	
77725	01267	00	00000	01750	
77726	01270	00	00000	23420	
77727	01271	00	00003	03240	
77730	01272	00	00036	41100	
77731	01273	00	00461	13200	
77732	01274	00	05753	60400	
77733	01275	00	73465	45000	
77734	01276	11	24027	62000	
77735	01277	00	00012	00000	
77736	01300	00	00000	00105	
77737	01301	00	00000	00017	

77740	01302	00	00000	00004
77741	01303	00	00000	00045
77742	01304	00	00000	00003
77743	01305	00	00000	00002
77744	01306	00	00001	00000
77745	01307	00	00001	00001
77746	01310	00	00000	00043
77747	01311	00	00000	00044
77750	01312	00	00000	00110
77751	01313	00	00005	00000

SPUR DIVISION AND GENERAL ALARMS

77752		75	30700	77754	STORE PART
77753		11	01100	75100	OF E.S.
77754		75	30021	01100	THIS TO
77755		11	77756	01100	E.S.
77756	01100	75	30200	01102	PART OF
77757	01101	11	77213	01200	CARD ROUTINE
77760	01102	31	01746	00000	ADDRESS
77761	01103	36	01310	10000	OF
77762	01104	55	10000	00030	INST.
77763	01105	11	01305	01371	5 DIGITS
77764	01106	61	00000	01302	CAR. RET.
77765	01107	37	01255	01250	ADDRESS
77766	01110	61	00000	01305	SPACE
77767	01111	31	01355	00052	PRINT
77770	01112	61	00000	20000	
77771	01113	34	20000	00006	LABEL
77772	01114	47	01112	01115	
77773	01115	61	00000	01311	DIVISOR
77774	01116	11	01766	10000	
77775	01117	37	01351	01346	DIVISOR MANTISSA
77776	01120	45	00000	01345	TO CONTINUE
77360	01345	55	01760	10003	INSTRUCTION
77361	01346	11	01335	01371	12 DIGITS
77362	01347	61	00000	01305	SPACE
77363	01350	37	01255	01250	Q OCTAL
77364	01351	37	01351	01352	
77365	01352	75	30700	76061	RESTORE
77366	01353	11	75100	01100	ES
77367	01354	56	00000	01741	EXCESS
77370	01355	22	14171	42403	FLEX CODE
76014		75	30021	76016	PART
76015		11	77756	01100	OF

FA (1960-01-100)

76016		75	30200	76022	DIVISION
76017		11	77213	01200	ALARM
76020		75	30700	76014	STORE
76021		11	01100	75100	ES
76022		75	30035	01137	THIS
76023		11	76024	01115	TO ES
76024	01115	61	00000	01142	PRINT LETTER
76025	01116	21	01115	01751	STEP
76026	01117	55	01760	10003	PRINT
76027	01120	37	01351	01346	REGISTER
76030	01121	21	01117	01151	STEP
76031	01122	37	01122	01123	SWITCH
76032	01123	61	00000	01302	CAR. RET.
76033	01124	37	01124	01125	SWITCH
76034	01125	37	01124	01115	PRINT N
76035	01126	21	01117	01150	STEP
76036	01127	37	01122	01115	PRINT
76037	01130	37	01124	01117	B
76040	01131	37	01131	01132	SWITCH
76041	01132	37	01131	01127	PRINT C
76042	01133	21	01117	01753	STEP
76043	01134	37	01131	01127	PRINT R
76044	01135	37	01131	01127	PRINT BOXES
76045	01136	45	00000	01352	
76046	01137	11	01141	01355	SET FLEX ALARM
76047	01140	45	00000	01102	
76050	01141	30	11301	20745	FLEX ALARM
76051	01142	00	00000	00014	
76052	01143	00	00000	00006	
76053	01144	00	00000	00023	
76054	01145	00	00000	00016	
76055	01146	00	00000	00012	

```
76056  01147  00 00000 00001
76057  01150  00 00004 00000
76060  01151  00 00001 00000
76061         37 01741 01741      SET NO ACCUMULATE
76062         37 01742 01742      SET STORE IN Y
76063         56 00000 01746      RETURN TO N. I.
```

SFUR SROK ALARM

72000	45 00000 72000	
72001	16 72026 72022	
72002	11 20000 00000	
72003	61 00000 72047	
72004	55 72012 00005	
72005	55 72012 00011	SET FOR 5 DIGITS
72006	55 72012 00012	SET FOR 2 DIGITS
72007	34 20000 00003	OCTAL
72010	32 72037 00000	
72011	11 20000 72012	PRINT
72012	00 01000 10001	
72013	44 72014 72007	LOOP
72014	11 10000 72012	RESTORE FLAG
72015	61 00000 72021	SPACE
72016	37 72016 72017	SWITCH
72017	31 00000 00044	
72020	11 72000 00000	SET JUMP
72021	37 72016 72004	PRINT
72022	37 72022 72023	SWITCH
72023	31 72000 00017	FOR MAIN
72024	15 20000 72025	ROUTINE AD
72025	16 72000 00000	SET JUMP
72026	15 72023 72025	RESTORE
72027	16 72027 72000	RESTORE
72030	31 72042 00047	
72031	37 72016 72047	PRINT ALARM
72032	41 00000 72033	ADJUST ADDRESS
72033	31 20000 00071	
72034	37 72016 72005	PRINT 5 DIGITS
72035	56 00000 00000	
72036	00 00000	EXCESS
72037	61 00000 72037	

72040	00 00000 52	
72041	00 00000 00 74	
72042	43 01130 12070	
72043	00 00000 00 64	
72044	00 00000 00062	
72045	00 00000 00 66	
72046	00 00000 00 72	
72047	61 00000 20045	FLEX
72050	34 20000 00006	PRINT
72051	47 72047 72015	LOOP

PX (1906-10-183)

SPUR PAPERTAPE DUMP

73234	37	73234	73235	EXIT
73235	45	00000	73265	AUTO FLEX
73236	45	00000	73270	AUTO BIOCTAL
73237	45	00000	73273	MANUAL FLEX
73240	45	00000	73277	MANUAL BIOCTAL
73241	11	00000	74000	
73242	11	73235	00000	
73243	75	30152	73245	
73244	11	00001	74001	
73245	75	30064	73247	
73246	11	73302	00006	
73247	11	10000	73300	
73250	11	20000	00151	
73251	54	20000	00044	
73252	11	20000	00152	
73253	37	73253	73254	
73254	11	73300	10000	
73255	31	00152	00044	
73256	32	00151	00000	
73257	11	00010	73275	
73260	11	00010	73300	
73261	75	30152	73263	RESTORE
73262	11	74001	00001	ES
73263	11	74000	00000	
73264	45	00000	73234	TO EXIT
73265	37	73253	73241	STORE IMAGE
73266	75	30057	00041	FLEX
73267	11	73366	00072	PROGRAM
73270	37	73253	73241	STORE IMAGE
73271	75	30040	00120	BIOCTAL
73272	11	73445	00072	PROGRAM
73273	11	10000	73275	PARAMETER WD.

73274		37	73234	73265	SET EXIT
73275		00	00000		P.W. STORAGE
73276		56	00000	73237	MANUAL FLEX.
73277		37	73234	73270	SET EXIT
73300		00	0000		Q
73301		56	00000	73240	MANUAL BIOCT.
73302	00006	00	00001	00000	
73303	00007	00	00000	00007	
73304	00010	00	00000	00000	
73305	00011	00	00000	00001	
73306	00012	00	00000	00002	
73307	00013	00	00000	00004	
73310	00014	00	00152	00000	
73311	00015	00	74000	00000	
73312	00016	63	00000	00017	
73313	00017	00	00000	00037	FLEX 0
73314	00020	00	00000	00052	
73315	00021	00	00000	00074	
73316	00022	00	00000	00070	
73317	00023	00	00000	00064	
73320	00024	00	00000	00062	
73321	00025	00	00000	00066	
73322	00026	00	00000	00072	
73323	00027	00	00000	00045	CAR. RET.
73324	00030	00	00000	00043	STOP CODE
73325	00031	00	00000	00075	
73326	00032	00	00000	00132	
73327	00033	00	00000	02000	
73330	00034	00	00000	40000	
73331	00035	11	00010	00001	CHECK ADDRESS
73332	00036	45	00000	00072	JUMP TO PUNCH
73333	00037	11	00034	00001	40000

```
73334   00040   45 00000 00072      JUMP TO PUNCH
73335   00041   75 00010 00043      LEADER
73336   00042   63 00000 00010
73337   00043   31 73234 00017      PARAMETER ADDRESS
73340   00044   11 20000 00001
73341   00045   31 00014 00000      PARAMETER
73342   00046   42 00001 00050       ADDRESS
73343   00047   21 00001 00015        IN IMAGE
73344   00050   15 00001 00051
73345   00051   11 30051 10000      PARAMETER
73346   00052   11 00010 00001      CLEAR
73347   00053   11 00010 00003       TEMPORARIES
73350   00054   15 10000 00003      AD. OF FIRST WD.
73351   00055   11 00003 00004
73352   00056   16 10000 00001      AD. OF LAST WD.
73353   00057   44 73254 00060
73354   00060   44 73254 00061      PAR. INST.
73355   00061   44 73254 00062
73356   00062   21 73234 00011      STEP
73357   00063   31 00001 00017      LAST AD.
73360   00064   34 00003 00071       FIRST AD.
73361   00065   11 20000 00002        NO. OF WDS.
73362   00066   46 00043 00067      PAR. END
73363   00067   21 00001 00011      STEP
73364   00070   43 00033 00035      2000 TO ZERO
73365   00071   43 00006 00037      10000 TO 40000
73366   00072   55 10000 00002
73367   00073   44 00074 00075      NO STOP PUNCH
73370   00074   16 00042 00135
73371   00075   31 00014 00000      CURRENT WORD
73372   00076   42 00004 00102       IN IMAGE
73373   00077   21 00003 00015      ADJUST FOR IMAGE
```

73374	00100	16	00031	00127	
73375	00101	45	00000	00103	
73376	00102	16	00101	00127	
73377	00103	11	00004	10000	CURRENT AD.
73400	00104	56	10000	00105	
73401	00105	55	10000	00006	
73402	00106	37	00150	00142	5 DIGITS
73403	00107	15	00003	00110	CURRENT
73404	00110	31	30110	00000	WORD
73405	00111	43	00010	00121	OMIT PUNCHING
73406	00112	75	00002	00114	2 SPACES
73407	00113	63	00000	00013	
73410	00114	11	20000	10000	CURRENT WD.
73411	00115	11	00011	00005	
73412	00116	37	00150	00143	2 DIGITS
73413	00117	37	00150	00141	5 DIGITS
73414	00120	37	00150	00141	5 DIGITS
73415	00121	41	00140	00124	32 WDS.
73416	00122	11	00017	00140	RESTORE TALLY
73417	00123	63	00000	00027	CAR.
73420	00124	63	00000	00027	RET.
73421	00125	21	00004	00006	STEP
73422	00126	11	00004	00003	
73423	00127	41	00002	30127	END
73424	00130	16	00032	00127	
73425	00131	11	00017	00002	
73426	00132	31	00140	00000	
73427	00133	43	00017	00135	
73430	00134	45	00000	00121	
73431	00135	63	00000	00030	
73432	00136	11	73431	00135	
73433	00137	45	00000	00041	TO NEXT PAR.

PX 71900-10-(183)

73434	00140	00	00000	00037	TALLY
73435	00141	63	00000	00013	SPACE
73436	00142	11	00013	00005	SET TALLY
73437	00143	55	10000	00003	
73440	00144	51	00007	20000	EXTRACT DIGIT
73441	00145	35	00016	00146	
73442	00146	00	00000	00000	PUNCH DIGIT
73443	00147	41	00005	00143	
73444	00150	45	00000	30150	EXIT
73445	00072	63	10000	00010	7TH LEVEL
73446	00073	11	00004	10000	PUNCH
73447	00074	55	10000	00025	INSERT
73450	00075	37	00130	00122	ADDRESS
73451	00076	37	00130	00122	
73452	00077	31	00014	00000	IN
73453	00100	42	00004	00104	IMAGE
73454	00101	21	00003	00015	ADJUST
73455	00102	16	00131	00113	
73456	00103	45	00000	00105	
73457	00104	16	00103	00113	
73460	00105	15	00003	00106	CURRENT ADDRESS
73461	00106	11	30000	10000	PUNCH
73462	00107	11	00013	00005	CURRENT
73463	00110	37	00130	00123	WORD
73464	00111	21	00004	00006	STEP
73465	00112	11	00004	00003	
73466	00113	41	00002	30113	END
73467	00114	11	00012	00005	
73470	00115	11	00001	10000	CHECK
73471	00116	37	00130	00123	ADDRESS
73472	00117	37	00130	00123	
73473	00120	75	00310	00041	PUNCH

73474	00121	63	00000	00010	LEADER
73475	00122	11	00011	00005	
73476	00123	55	10000	00006	PUNCH
73477	00124	63	00000	10000	2
73500	00125	41	00005	00123	DIGITS
73501	00126	55	10000	00006	
73502	00127	63	10000	10000	
73503	00130	56	10000	30130	
73504	00131	00	00000	00077	

PX 71900-10-(183)

SUPER OTTO SPUR

72052	01000	45	00000	01013
72053	01001	47	01002	01004
72054	01002	35	01304	01172
72055	01003	45	00000	01171
72056	01004	55	00022	00043
72057	01005	53	10000	00004
72060	01006	53	10000	00013
72061	01007	45	00000	01173
72062	01010	56	00000	01010
72063	01011	11	01316	10000
72064	01012	37	01273	01266
72065	01013	45	30000	01016
72066	01014	75	00002	01020
72067	01015	17	00000	01332
72070	01016	75	00276	01020
72071	01017	63	00000	01314
72072	01020	45	10000	01042
72073	01021	11	01317	10000
72074	01022	37	01273	01266
72075	01023	11	01320	10000
72076	01024	37	01273	01267
72077	01025	11	40000	00003
72100	01026	11	01275	40000
72101	01027	57	00000	00000
72102	01030	11	00003	40000
72103	01031	15	01332	01033
72104	01032	15	01333	01034
72105	01033	11	30000	10000
72106	01034	31	30000	00000
72107	01035	56	10000	01036
72110	01036	11	10000	00017

72111	01037	11	20000	00020
72112	01040	37	01261	01262
72113	01041	44	72441	01044
72114	01042	23	10000	10000
72115	01043	56	00000	01036
72116	01044	61	00000	01340
72117	01045	11	01334	00023
72120	01046	11	00020	10000
72121	01047	61	00000	01340
72122	01050	37	01261	01264
72123	01051	11	00017	10000
72124	01052	51	01355	20000
72125	01053	73	01345	00003
72126	01054	36	10000	00025
72127	01055	11	00003	20000
72130	01056	42	01335	01070
72131	01057	11	00017	00026
72132	01060	11	01276	01110
72133	01061	11	01312	01202
72134	01062	55	00017	10005
72135	01063	44	01064	01075
72136	01064	11	01277	01114
72137	01065	11	01300	01210
72140	01066	75	20002	01101
72141	01067	55	00026	00006
72142	01070	11	00017	00027
72143	01071	21	00017	01336
72144	01072	11	01301	01110
72145	01073	11	01302	01202
72146	01074	45	00000	01076
72147	01075	11	00020	00027

PX 71900-10-(183)

72150	01076	11	01303	01114
72151	01077	11	01313	01210
72152	01100	45	00000	01066
72153	01101	45	30000	01175
72154	01102	15	00017	01107
72155	01103	75	30014	01105
72156	01104	23	00003	00003
72157	01105	11	01311	00000
72160	01106	17	00000	01333
72161	01107	11	30000	00030
72162	01110	11	00026	30000
72163	01111	11	01337	00022
72164	01112	11	01340	00023
72165	01113	37	01174	01166
72166	01114	11	00027	30000
72167	01115	11	01341	00022
72170	01116	11	01340	00023
72171	01117	37	01174	01166
72172	01120	45	20000	01152
72173	01121	11	00030	00021
72174	01122	11	01342	00022
72175	01123	11	01337	00023
72176	01124	37	01174	01166
72177	01125	11	01337	00024
72200	01126	55	00022	00043
72201	01127	11	01340	00023
72202	01130	37	01174	01166
72203	01131	41	00024	01126
72204	01132	11	01343	00024
72205	01133	16	01305	01135
72206	01134	77	00000	01314

72207	01135	77	10000	30000
72210	01136	77	10000	01314
72211	01137	23	01135	01337
72212	01140	41	00024	01134
72213	01141	56	20000	01142
72214	01142	41	00025	01147
72215	01143	17	00000	01332
72216	01144	45	10000	01043
72217	01145	75	20002	01033
72220	01146	21	01033	01344
72221	01147	21	01107	01345
72222	01150	75	20002	01103
72223	01151	21	00026	01346
72224	01152	11	00030	10000
72225	01153	51	01347	20000
72226	01154	43	01350	01156
72227	01155	45	00000	01121
72230	01156	11	00030	00021
72231	01157	11	01342	00022
72232	01160	11	01351	00024
72233	01161	11	01352	00023
72234	01162	37	01174	01166
72235	01163	55	00022	00043
72236	01164	41	00024	01161
72237	01165	45	00000	01132
72240	01166	55	00021	00003
72241	01167	51	01353	20000
72242	01170	45	00000	01001
72243	01171	55	00022	00043
72244	01172	53	01321	30000
72245	01173	41	00023	01166

72246	01174	45	00000	01174
72247	01175	15	00017	01200
72250	01176	75	00036	01200
72251	01177	63	00000	01314
72252	01200	11	30000	00030
72253	01201	63	00000	01307
72254	01202	11	00026	30000
72255	01203	37	01261	01250
72256	01204	75	00002	01210
72257	01205	63	00000	01340
72260	01206	75	00007	01210
72261	01207	63	00000	01340
72262	01210	11	00027	30000
72263	01211	37	01261	01250
72264	01212	75	00002	01216
72265	01213	63	00000	01340
72266	01214	75	00007	01216
72267	01215	63	00000	01340
72270	01216	45	20000	01235
72271	01217	11	00030	10000
72272	01220	11	01337	00023
72273	01221	37	01261	01253
72274	01222	63	00000	01340
72275	01223	37	01261	01250
72276	01224	63	00000	01340
72277	01225	37	01261	01250
72300	01226	56	20000	01227
72301	01227	41	00025	01232
72302	01230	75	00003	01144
72303	01231	63	00000	01307
72304	01232	21	01200	01345

72305	01233	75	20002	01200
72306	01234	21	00026	01346
72307	01235	11	00030	10000
72310	01236	51	01347	20000
72311	01237	43	01350	01241
72312	01240	45	00000	01217
72313	01241	11	01351	00024
72314	01242	11	00030	10000
72315	01243	11	01352	00023
72316	01244	37	01261	01253
72317	01245	63	00000	01340
72320	01246	41	00024	01243
72321	01247	45	00000	01226
72322	01250	11	01340	00023
72323	01251	45	00000	01253
72324	01252	11	01343	00023
72325	01253	15	01305	01256
72326	01254	55	10000	00003
72327	01255	51	01353	20000
72330	01256	35	30000	01257
72331	01257	63	00000	30000
72332	01260	41	00023	01254
72333	01261	45	00000	01261
72334	01262	61	00000	01307
72335	01263	11	01343	00023
72336	01264	15	01307	01256
72337	01265	45	00000	01254
72340	01266	61	00000	01307
72341	01267	11	01354	00023
72342	01270	55	10000	00006
72343	01271	61	00000	10000

72344	01272	41	00023	01270
72345	01273	45	00000	01273
72346	01274	45	00000	01013
72347	01275	45	00000	01030
72350	01276	11	00026	00021
72351	01277	45	00000	01120
72352	01300	45	00000	01214
72353	01301	45	00000	01114
72354	01302	45	00000	01206
72355	01303	11	00027	00021
72356	01304	53	01321	00005
72357	01305	00	01306	00016
72360	01306	63	00000	01322
72361	01307	00	01310	00045
72362	01310	61	00000	01322
72363	01311	45	00000	01103
72364	01312	11	00026	10000
72365	01313	11	00027	10000
72366	01314	00	00000	00000
72367	01315	24	34070	40613
72370	01316	03	01030	40336
72371	01317	15	30123	00720
72372	01320	01	12240	41406
72373	01321	77	77777	77777
72374	01322	00	00000	00037
72375	01323	00	00000	00052
72376	01324	00	00000	00074
72377	01325	00	00000	00070
72400	01326	00	00000	00064
72401	01327	00	00000	00062
72402	01330	00	00000	00066

72403	01331	00	00000	00072
72404	01332	00	00300	00110
72405	01333	00	00301	00112
72406	01334	00	00000	00006
72407	01335	00	00000	20000
72410	01336	00	74000	00000
72411	01337	00	00000	00001
72412	01340	00	00000	00004
72413	01341	00	40000	00000
72414	01342	00	00200	00000
72415	01343	00	00000	00013
72416	01344	00	00002	00000
72417	01345	00	00001	00000
72420	01346	00	00100	00000
72421	01347	77	00000	00000
72422	01350	14	00000	00000
72423	01351	00	00000	00002
72424	01352	00	00000	00003
72425	01353	00	00000	00007
72426	01354	00	00000	00005
72427	01355	00	77777	77777
72430	01356	00	00000	00000
72431		11	00000	74000
72432		11	72540	00000
72433		75	31777	72435
72434		11	00001	74001
72435		75	30260	72437
72436		11	72445	00020
72437		75	30073	00020
72440		11	72725	00700
72441		75	31777	72443

PX 71900-10-(183)

72442		11	74001	00001
72443		11	74000	00000
72444		57	00000	00000
72445	00020	17	00000	00700
72446	00021	71	00704	00711
72447	00022	11	20000	00014
72450	00023	55	10000	00002
72451	00024	11	10000	00015
72452	00025	55	10000	00001
72453	00026	11	10000	00016
72454	00027	11	10000	00017
72455	00030	21	00017	00015
72456	00031	11	00703	00011
72457	00032	75	30010	00250
72460	00033	23	00001	00001
72461	00034	45	10000	00216
72462	00035	11	00714	10000
72463	00036	37	00227	00223
72464	00037	11	00715	10000
72465	00040	37	00227	00223
72466	00041	11	00716	10000
72467	00042	37	00227	00223
72470	00043	11	40000	00010
72471	00044	11	00747	00000
72472	00045	11	00747	40000
72473	00046	57	00300	00000
72474	00047	17	00301	00702
72475	00050	11	00010	40000
72476	00051	15	00046	00053
72477	00052	15	00047	00054
72500	00053	11	30300	10000

72501	00054	31	30301	00000
72502	00055	56	10000	00056
72503	00056	11	10000	00001
72504	00057	11	20000	00002
72505	00060	61	00000	00716
72506	00061	61	00000	00716
72507	00062	11	00001	10000
72510	00063	11	00707	00007
72511	00064	37	00237	00232
72512	00065	37	00237	00230
72513	00066	37	00237	00230
72514	00067	44	00220	00070
72515	00070	11	00002	10000
72516	00071	61	00000	00716
72517	00072	61	00000	00727
72520	00073	61	00000	00727
72521	00074	55	10000	00005
72522	00075	37	00237	00230
72523	00076	61	00000	00712
72524	00077	75	00005	00101
72525	00100	61	00000	00727
72526	00101	15	00001	00131
72527	00102	11	00001	10000
72530	00103	55	10000	00003
72531	00104	51	00731	00003
72532	00105	55	10000	00017
72533	00106	51	00731	00004
72534	00107	11	00732	20000
72535	00110	42	00003	00114
72536	00111	21	00131	00733
72537	00112	11	00124	00136

PX 71900-10-(183)

72540	00113	45	00000	00130
72541	00114	11	00734	10000
72542	00115	51	00001	20000
72543	00116	47	00127	00117
72544	00117	11	00002	10000
72545	00120	55	10000	00003
72546	00121	51	00731	00005
72547	00122	11	00125	00136
72550	00123	45	00000	00130
72551	00124	45	00000	00153
72552	00125	45	00000	00137
72553	00126	45	00000	00146
72554	00127	11	00126	00136
72555	00130	75	31000	00132
72556	00131	11	30000	01000
72557	00132	21	00131	00735
72560	00133	15	00735	00134
72561	00134	11	31000	00006
72562	00135	21	00134	00736
72563	00136	45	00000	30136
72564	00137	11	00003	00010
72565	00140	15	00703	00011
72566	00141	37	00247	00240
72567	00142	21	00011	00014
72570	00143	11	00005	00010
72571	00144	37	00247	00240
72572	00145	45	00000	00157
72573	00146	11	00003	00010
72574	00147	15	00703	00011
72575	00150	37	00247	00240
72576	00151	21	00011	00017

72577	00152	45	00000	00160
72600	00153	15	00703	00011
72601	00154	21	00011	00016
72602	00155	11	00003	00010
72603	00156	37	00247	00240
72604	00157	21	00011	00015
72605	00160	45	20000	00256
72606	00161	11	00006	00010
72607	00162	55	00010	00041
72610	00163	11	00707	00007
72611	00164	37	00247	00241
72612	00165	21	00011	00704
72613	00166	11	00006	00010
72614	00167	55	00010	00003
72615	00170	37	00247	00240
72616	00171	21	00011	00704
72617	00172	37	00247	00240
72620	00173	11	00740	10000
72621	00174	51	00011	20000
72622	00175	43	00013	00206
72623	00176	21	00011	00705
72624	00177	11	00003	20000
72625	00200	43	00004	00213
72626	00201	21	00003	00741
72627	00202	21	00005	00741
72630	00203	11	00134	20000
72631	00204	43	00742	00130
72632	00205	45	00000	00134
72633	00206	16	00703	00011
72634	00207	17	00000	00701
72635	00210	17	00000	00700

PX 71900-10-(183)

72636	00211	17	00000	00702
72637	00212	45	00000	00177
72640	00213	45	10000	00216
72641	00214	75	20002	00053
72642	00215	21	00053	00743
72643	00216	23	10000	10000
72644	00217	56	00000	00056
72645	00220	17	00000	00701
72646	00221	75	00003	72441
72647	00222	17	00000	00700
72650	00223	11	00713	00007
72651	00224	55	10000	00006
72652	00225	61	00000	10000
72653	00226	41	00007	00224
72654	00227	45	00000	30227
72655	00230	61	00000	00712
72656	00231	11	00712	00007
72657	00232	55	10000	00003
72660	00233	51	00744	20000
72661	00234	35	00730	00235
72662	00235	61	00000	30235
72663	00236	41	00007	00232
72664	00237	45	00000	30237
72665	00240	11	00712	00007
72666	00241	11	00737	10000
72667	00242	53	00010	00011
72670	00243	45	00000	00762
72671	00244	21	00011	00704
72672	00245	54	00010	00003
72673	00246	41	00007	00242
72674	00247	45	00000	30247

72675	00250	11	00740	10000
72676	00251	51	00703	20000
72677	00252	72	00705	00706
72700	00253	11	20000	00013
72701	00254	17	00000	00702
72702	00255	45	00000	00034
72703	00256	11	00006	10000
72704	00257	51	00745	20000
72705	00260	43	00746	00262
72706	00261	45	00000	00161
72707	00262	11	00006	00010
72710	00263	55	00010	00041
72711	00264	11	00711	00007
72712	00265	37	00247	00241
72713	00266	21	00011	00704
72714	00267	11	00006	00010
72715	00270	55	00010	00011
72716	00271	11	00711	00007
72717	00272	37	00247	00241
72720	00273	21	00011	00704
72721	00274	11	00711	00007
72722	00275	37	00247	00241
72723	00276	45	00000	00173
72724	00277	00	00000	00000
72725	00700	00	00000	06000
72726	00701	00	00000	04400
72727	00702	00	00000	05000
72730	00703	00	00200	00020
72731	00704	00	00024	00000
72732	00705	00	00000	00040
72733	00706	00	00000	00037

PX 71900-10-(183)

72734	00707	00	00000	00001
72735	00710	00	00000	00002
72736	00711	00	00000	00003
72737	00712	00	00000	00004
72740	00713	00	00000	00005
72741	00714	45	47150	10411
72742	00715	03	30220	40407
72743	00716	22	04240	15745
72744	00717	00	00000	00037
72745	00720	00	00000	00052
72746	00721	00	00000	00074
72747	00722	00	00000	00070
72750	00723	00	00000	00064
72751	00724	00	00000	00062
72752	00725	00	00000	00066
72753	00726	00	00000	00072
72754	00727	00	00000	00042
72755	00730	61	00000	00717
72756	00731	07	77770	00000
72757	00732	00	17770	00000
72760	00733	00	74000	00000
72761	00734	01	00000	00000
72762	00735	00	01000	00000
72763	00736	00	00001	00000
72764	00737	07	00000	00000
72765	00740	00	00000	77777
72766	00741	00	00010	00000
72767	00742	11	02000	00006
72770	00743	00	00002	00000
72771	00744	00	00000	00007
72772	00745	77	00000	00000

72773	00746	14	00000	00000
72774	00747	45	00000	00047
72775	00750	00	00000	01777
72776	00751	11	73015	72461
72777	00752	45	00000	72431
73000	00753	45	00000	72776
73001	00754	45	00000	72431
73002	00755	45	00000	73020
73003	00756	11	00771	76461
73004	00757	11	00750	00001
73005	00760	16	00772	00200
73006	00761	45	00000	00101
73007	00762	42	00766	00764
73010	00763	35	00767	20000
73011	00764	77	10000	20000
73012	00765	45	00000	00244
73013	00766	06	00000	00000
73014	00767	02	00000	00000
73015	00770	45	00000	00756
73016	00771	45	10000	00216
73017	00772	00	00000	00220
73020		11	00000	74000
73021		11	72152	00000
73022		75	31777	73024
73023		11	00001	74001
73024		75	30357	01013
73025		11	72052	01000

PX 71900-10-(183)

SPUR SUBROUTINE ASSEMBLY

76151		15	01601	76153	Y ADDRESS
76152		75	30052	76154	PW FOR
76153		11	30000	00002	EXAMINATION
76154		75	30304	00056	THIS TO
76155		11	76156	00056	ES
76156	00056	15	01753	00121	PW ACQUISITION
76157	00057	11	00347	00372	INITIAL INST. LIMIT
76160	00060	11	01752	00373	INITIAL TEMP. LIMIT
76161	00061	15	01753	00064	SET TREE TEST
76162	00062	31	01761	00000	
76163	00063	36	01751	00374	SET TALLY
76164	00064	11	30000	20000	
76165	00065	42	00357	00067	WHIFFLETREE
76166	00066	42	00360	00115	TEST
76167	00067	21	00064	01753	STEP
76170	00070	41	00374	00064	
76171	00071	41	01761	00120	TEST FOR END
76172	00072	75	31240	00074	STORE
76173	00073	11	00400	74400	MODIFIED ES
76174	00074	75	30200	00076	PART OF
76175	00075	11	77213	01200	CARD ROUTINE
76176	00076	61	00000	01302	CAR. RET.
76177	00077	31	00361	00052	PRINT
76200	00100	61	00000	20000	AVAIL.
76201	00101	34	20000	00006	AND
76202	00102	47	00100	00103	ES
76203	00103	55	00373	10030	FREE
76204	00104	61	00000	01305	SPACE
76205	00105	11	01305	01371	LIMITS
76206	00106	37	01255	01250	
76207	00107	37	00107	00110	

76210	00110	23	00372	01751	
76211	00111	55	20000	00030	
76212	00112	37	00107	00104	
76213	00113	75	31777	01746	EXIT
76214	00114	11	74001	00001	RESTORE E.S.
76215	00115	23	00372	00322	ADJUST INST. LIMITS
76216	00116	75	30007	00071	WHIFFLETREE
76217	00117	11	00350	01472	ADDITION
76220	00120	75	30002	00122	CURRENT
76221	00121	11	30000	00362	P W
76222	00122	21	00121	01753	STEP
76223	00123	15	00362	00165	SET SUB. R. EXTRACT.
76224	00124	31	00363	00060	
76225	00125	11	20000	00364	NUMBER OF INST.
76226	00126	34	20000	00014	
76227	00127	11	20000	00365	NUMBER OF CONST.
76230	00130	34	20000	00014	
76231	00131	11	20000	00366	NUMBER OF TEMP.
76232	00132	31	00364	00000	NUMBER
76233	00133	35	00365	00374	INST. AND CONST.
76234	00134	35	00317	00164	SUB. R.
76235	00135	55	00164	00017	TRANSFER
76236	00136	23	00372	00374	NEW INST. LIMIT
76237	00137	42	00343	00157	ALARM
76240	00140	16	20000	00165	SUB. R. LANDING
76241	00141	16	20000	00312	INST. RETURN
76242	00142	36	00346	00367	INST. MODIFICATION
76243	00143	11	00340	10000	
76244	00144	52	00362	00363	MODIFIED ENTRANCE
76245	00145	13	00341	20000	
76246	00146	36	00374	00370	TEMP. MOD.

```
76247  00147  55 00362 10006
76250  00150  51 00336 00362    O P
76251  00151  31 00366 00000     TEST
76252  00152  35 01752 20000      TEMP
76253  00153  42 00373 00155       LIMIT
76254  00154  11 20000 00373    CHANGE LIMIT
76255  00155  11 00373 20000
76256  00156  42 00372 00161
76257  00157  75 31777 76020    ALARM
76260  00160  11 74001 00001
76261  00161  31 00346 00000
76262  00162  35 00374 00365    UPPER LIMIT INST. MOD.
76263  00163  35 00366 00366    UPPER LIMIT TEMP.
76264  00164  30 30000 30000     SUB. R. FOR
76265  00165  11 30000 30000      MODIFICATION
76266  00166  37 00260 00251     MODIFY
76267  00167  45 00000 00312      U ONLY
76270  00170  31 00165 00017     SET INITIAL
76271  00171  15 20000 00233      INST ACQUISITION
76272  00172  41 00364 00233    TALLY INST. MOD.
76273  00173  11 00315 00231
76274  00174  31 00363 00017
76275  00175  35 00363 00363
76276  00176  55 00362 10043
76277  00177  44 00201 00200
76300  00200  11 00316 00231
76301  00201  55 00362 10041
76302  00202  51 00334 20000
76303  00203  55 10000 00003
76304  00204  43 00322 00222    TEST  60 TO 67
76305  00205  16 00344 00231    SET FOR  10,30
```

76306	00206	43	00320	00220	TEST FOR
76307	00207	43	01751	00220	10,30
76310	00210	43	00334	00212	TEST FOR 70 TO 77
76311	00211	45	00000	00157	
76312	00212	16	00345	00231	SET FOR 74
76313	00213	51	00334	20000	
76314	00214	43	00321	00231	TEST FOR 74
76315	00215	16	00332	00231	SET FOR 70, 71, 72, 73
76316	00216	42	00321	00222	TEST FOR 75, 76, 77
76317	00217	45	00000	00157	ALARM
76320	00220	51	00324	20000	TEST FOR
76321	00221	43	00324	00157	17, 37
76322	00222	51	00322	10000	
76323	00223	47	00224	00231	
76324	00224	43	01752	00230	
76325	00225	43	00321	00227	
76326	00226	23	00231	01751	
76327	00227	23	00231	01752	
76330	00230	23	00231	01751	
76331	00231	30	00000	00000	
76332	00232	45	00000	00071	
76333	00233	11	30000	00375	CURRENT INST.
76334	00234	21	00233	00337	STEP
76335	00235	55	00375	10006	
76336	00236	51	00336	20000	
76337	00237	43	00323	00271	14 SPECIAL
76340	00240	43	00324	00261	V ONLY
76341	00241	43	00327	00261	V ONLY
76342	00242	42	00325	00251	U, V
76343	00243	42	00326	00166	U ONLY
76344	00244	42	00330	00251	U, V

PX 71900-10-(183)

```
76345   00245   42 00331 00166      U ONLY
76346   00246   42 00333 00261      V ONLY
76347   00247   42 00335 00251      U, V
76350   00250   45 00000 00261      V ONLY
76351   00251   55 00375 00025
76352   00252   51 00340 10000
76353   00253   42 00346 00256      NO MOD.
76354   00254   42 00365 00265      MOD INST
76355   00255   42 00366 00267      MOD TEMP
76356   00256   37 00256 00257
76357   00257   55 00375 00017
76360   00260   37 00260 00261
76361   00261   11 00375 10000
76362   00262   37 00256 00252
76363   00263   37 00263 00264
76364   00264   45 00000 00312      TO RESTOR
76365   00265   21 00375 00367
76366   00266   45 00000 00256
76367   00267   21 00375 00370
76370   00270   45 00000 00256
76371   00271   11 00375 00371      DUPLICATE
76372   00272   11 00375 10000       SYNTHETIC
76373   00273   51 01757 00375
76374   00274   51 00342 20000
76375   00275   31 20000 00003
76376   00276   35 00375 00375              INSTRUCTION
76377   00277   55 00371 10006
76400   00300   44 00301 00303      ONE, TWO ADDRESS
76401   00301   37 00263 00261
76402   00302   45 00000 00304
76403   00303   37 00263 00251
```

76404	00304	55	00375	20041	REPLACE
76405	00305	11	00342	10000	
76406	00306	53	20000	00371	WITH
76407	00307	11	01757	10000	SYNTHETIC
76410	00310	53	00375	00371	
76411	00311	11	00371	00375	PARTS
76412	00312	11	00375	30000	RETURN INST.
76413	00313	21	00312	01751	STEP
76414	00314	45	00000	00172	
76415	00315	15	00363	01477	
76416	00316	16	00363	01477	
76417	00317	00	17075	30000	
76420	00320	00	00000	00003	
76421	00321	00	00000	00004	
76422	00322	00	00000	00006	
76423	00323	00	00000	00014	
76424	00324	00	00000	00017	
76425	00325	00	00000	00031	
76426	00326	00	00000	00035	
76427	00327	00	00000	00045	
76430	00330	00	00000	00054	
76431	00331	00	00000	00056	
76432	00332	00	00000	01505	
76433	00333	00	00000	00071	
76434	00334	00	00000	00007	
76435	00335	00	00000	00075	
76436	00336	00	00000	00077	
76437	00337	00	00001	00000	
76440	00340	00	00000	77777	
76441	00341	00	00000	00776	
76442	00342	00	00177	70000	

76443	00343	00	00000	00400	LOWE LIMIT INST
76444	00344	00	00000	01632	
76445	00345	00	00000	01531	
76446	00346	00	00000	01000	
76447	00347	00	00000	01500	
76450	00350	44	01473	01474	01472
76451	00351	44	30067	30066	01473
76452	00352	44	30065	30064	01474
76453	00353	44	01476	01477	01475
76454	00354	44	30063	30062	01476
76455	00355	44	30061	30060	01477
76456	00356	44	01472	01475	01500
76457	00357	60	00000	00000	WHIFFLETREE
76460	00360	67	77777	77777	EXTENSION TESTS
76461	00361	30	17301	41142	

SPUR SQUARE ROOT

70252	01000	11	01766	01774	CASE ZERO
70253	01001	11	01766	20000	N TO A
70254	01002	46	76020	01003	NEGATIVE
70255	01003	47	01004	01742	ZERO
70256	01004	12	01767	10000	P EVEN
70257	01005	55	10000	00043	OR
70260	01006	44	01007	01011	ODD
70261	01007	21	01767	01751	P 1 TO P
70262	01010	54	01766	00107	N/2 TO N
70263	01011	55	01767	00043	N /2 TO P
70264	01012	11	01767	01775	P TO R
70265	01013	43	01025	01742	N EQUAL .9
70266	01014	11	01025	01762	.9 2 TO X
70267	01015	31	01766	00042	COMPUTE
70270	01016	73	01762	01763	
70271	01017	54	01762	00107	SQUARE
70272	01020	23	10000	01762	
70273	01021	21	01762	01763	
70274	01022	44	01015	01023	ROOT
70275	01023	11	20000	01774	N TO R
70276	01024	45	00000	01742	EXT
70277	01025	37	77777	77777	.9

SPUR CUBE ROOT

70300	01000	11	01766	01774	CASE ZERO
70301	01001	11	01766	20000	N TO A
70302	01002	47	01003	01742	N ZERO
70303	01003	11	01767	20000	P/3
70304	01004	73	01036	10000	TO Q
70305	01005	47	01006	01013	REM. ZERO
70306	01006	43	01752	01011	REM. 2
70307	01007	21	01767	01751	P 1 TO &
70310	01010	54	01766	00107	N / 2 TO N
70311	01011	21	01767	01751	P 1 TO P
70312	01012	54	01766	00107	N / 2 TO N
70313	01013	11	01767	20000	P / 3
70314	01014	73	01036	01775	TP P
70315	01015	11	01037	01762	.9 X 2 TO X
70316	01016	11	01766	20000	N
70317	01017	46	01020	01021	NEG.
70320	01020	13	01037	01762	-.9 TO X
70321	01021	43	01762	01742	
70322	01022	71	01762	01762	X
70323	01023	54	20000	00045	EXP 35
70324	01024	11	20000	01763	
70325	01025	54	01766	20043	N EXP 70
70326	01026	73	01763	20000	X
70327	01027	36	01762	20000	-X EXP 35
70330	01030	73	01036	10000	X
70331	01031	21	01762	10000	X EXP &5
70332	01032	71	10000	01766	N X
70333	01033	46	01022	01034	NEG.
70334	01034	11	01762	01774	X EQUAL N
70335	01035	45	00000	01742	EXIT
70336	01036	00	00000	00003	
70337	01037	37	77777	77777	.9

SPUR nth ROOT SUBROUTINE

70200	01000	11	01761	20000	$n \rightarrow$ A
70201	01001	42	01752	76020	$n < 2 \rightarrow$ ALARM
70202	01002	75	30004	01004	N $\rightarrow$ S, R
70203	01003	11	01770	01772	
70204	01004	11	01770	20000	N $\rightarrow$ A
70205	01005	47	01006	01736	N = 0 ? $\rightarrow$ EXIT
70206	01006	11	01051	01764	.9 $\rightarrow$ P
70207	01007	46	01010	01013	N NEG. ?
70210	01010	13	01764	01764	−.9 $\rightarrow$ P
70211	01011	55	01761	10043	n EVEN $\rightarrow$ ALARM
70212	01012	44	01013	76020	
70213	01013	11	01771	20000	EXP $\rightarrow$ A
70214	01014	73	01761	01775	EXP/N $\rightarrow$ 1775, REM. $\rightarrow$ A
70215	01015	11	10000	01765	EXP/N $\rightarrow$ P
70216	01016	47	01017	01022	REM. = 0 ?
70217	01017	21	01765	01751	EXP/N + 1 $\rightarrow$ P
70220	01020	75	30002	01024	S = N $\rightarrow$ C
70221	01021	11	01772	01770	
70222	01022	11	01770	20000	N $\rightarrow$ A
70223	01023	43	01764	01742	N = ±.9 $\rightarrow$ EXIT
70224	01024	11	01761	20000	$n \rightarrow$ A
70225	01025	36	01752	01052	$n - 2 = J \rightarrow 1052$
70226	01026	11	01764	01766	$X_{i-1} \rightarrow$ B
70227	01027	11	01765	01767	
70230	01030	37	01030	01031	
70231	01031	37	01741	01633	N/X_{i-1}^{n-1}
70232	01032	41	01052	01026	
70233	01033	37	01030	01026	$X_{i-1} \rightarrow$ B
70234	01034	37	01741	01655	$N/X_{i-1}^{n-1} - X_{i-1}$
70235	01035	73	01761	20000	$/n = \Delta X$
70236	01036	37	01741	01712	NORMALIZE Δ X
70237	01037	47	01040	01047	Δ X = 0 ? $\rightarrow$ EXIT

PX 71900-10-(183)

70240	01040	11	01775	01052	ΔX EXP. → 1052
70241	01041	37	01030	01026	X_{i-1} → B
70242	01042	37	01741	01656	$X_{i-1} + \Delta X = X_i$
70243	01043	11	01774	01764	} X_i → P
70244	01044	11	01775	01765	}
70245	01045	23	01771	01052	(X_i EXP) - (ΔX EXP) → A
70246	01046	42	01754	01020	A > 35 → EXIT
70247	01047	75	30002	01742	EXIT
70250	01050	11	01764	01774	X_i → R
70251	01051	37	77777	77777	.9

THE RAMO-WOOLDRIDGE CORPORATION
LOS ANGELES 45, CALIFORNIA

DATE: January 9, 1956

SUBJECT: Computing Center Organization

FROM: W. F. Bauer

The following delineates the areas of interest, cognizance, and responsibility of the five groups comprising the organization of the Digital Computing Center.

SYSTEMS AND TRAINING

1. Develop, maintain and improve a comprehensive computation system for the 1103 and 1103A including component parts such as compiler, algebraic coding scheme, service routines, and the routines necessary for effecting an automatic system, implementing, insofar as is possible, the suggestions for such programs and routines made by all other groups.

2. Develop such administrative procedures concomitant to the computation system to ensure the proper flow of information to and from the personnel of the Operations Group which operates the various equipments, and cooperate with that group in effecting the plan.

3. Keep subroutine and service routine library records, disseminate appropriate programming and system operation information to Computing Center personnel and other interested parties, and participate in cooperative programming efforts and in the exchange of program information between 1103 or 1103A users.

4. Aid in the general recruiting program and assume major responsibility in the initial instruction and training of personnel in programming techniques.

5. Recruit and train computer operators, operate the 1103 (or 1103A) computer, and develop operating techniques necessary to insure the highest standards of computer use.

6. Carry out specific projects in programming and computer studies as may be assigned.

OPERATIONS

1. Aid the Systems Group in developing methods and procedures for operation of the 1103 (or 1103A) computation system, and appropriately execute such procedures.

PX 71900-10-(10:27)

2. Schedule 1103 (or 1103A) computer time and develop and initiate methods and procedures as necessary to generate and maintain adequate records concerning problem assignments, records necessary for budget and fiscal requirements, and records of computer operating data.

3. Operate all auxiliary and off-line equipment, including hand computer equipment, recruit and train operators of such equipment, and design procedures as are necessary in the operation of such equipment.

4. Determine the need and specifications for any equipment to be modified, purchased or especially built, and transmit information as needed to programming groups on the operation and characteristics of such equipment.

5. Perform computer programming as necessary, especially on those problems involving special equipment such as analog-digital converters used in simulation and data reduction activities.

PROGRAMMING

1. Make contacts as necessary with customers and problem originators in the programming and running of assigned problems.

2. Assume major responsibility for the central GMRD programming effort as assigned, such as aerodynamics, structures, and air frame and trajectory analysis.

3. Cooperate with and advise the Numerical Analysis and Systems Groups on the development of routines necessary for production computation.

4. Cooperate with and advise the Operations Group in the development of a computation system and its subsequent adoption.

5. Assume major responsibility for the general recruiting program and the evaluation of prospective programmers.

NUMERICAL ANALYSIS

1. Consult with customers or problem originators on problems requiring special attention in regards formulation for numerical handling, advise on the formulation and machine plan of all problems in the Computing Center, and assume major responsibility for certain problems as assigned.

2. Conduct basic and applied research in numerical analysis and machine techniques and publish the results whenever appropriate.

3. Ascertain the need for, determine specifications, and develop techniques of standard numerical operations (e.g. matrix inversion, differential equation solving) as are needed and, when appropriate, perform the computer programming necessary.

4. Aid in the general recruiting program, and assume major responsibility in the recruitment of numerical analysts in cooperation with the Programming Group.

5. Provide technical support and coordination to the activities of consultants to the Computing Center in numerical analysis.

MATHEMATICAL ANALYSIS

1. Consult with customers or problem originators on problems, performing such mathematical research and analysis as is assigned or necessary; advise the Computing Center on the formulation of problems when appropriate; and consult with the Numerical Analysis Group to determine the combination of machine and analytical techniques likely to give most fruitful results.

2. Conduct basic research in Applied Mathematics as is appropriate and publish such results.

3. Aid in the general recruiting program, and assume major responsibility in the recruitment of applied mathematicians in cooperation with the Programming Group.

4. Provide technical support and coordination to the activities of consultants to the Computing Center in applied mathematics.

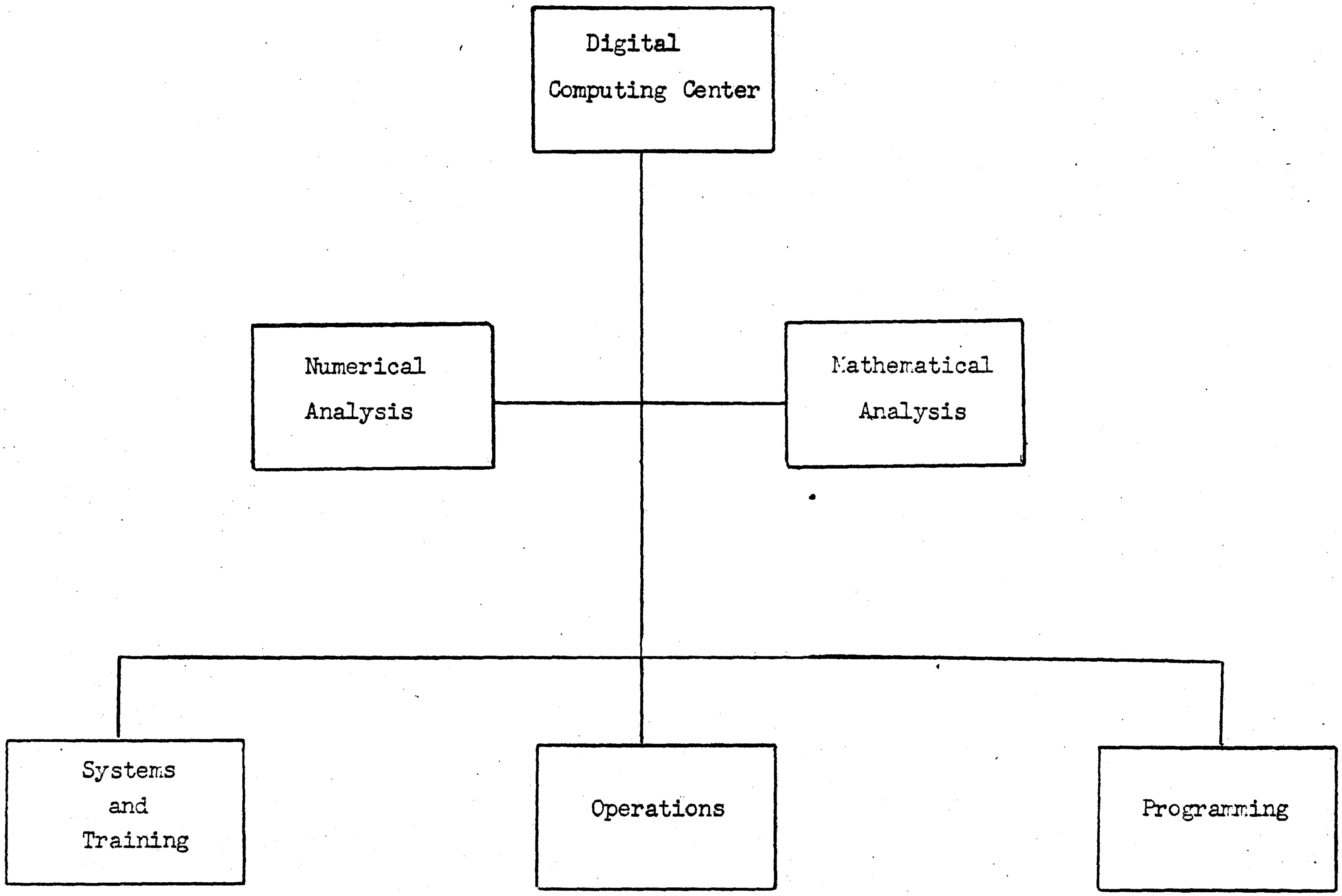
5. Provide and maintain sources of information in applied mathematics in the form of appropriate seminars, lectures, courses, and the Computing Center Library.

Walter F. Bauer

Walter F. Bauer

:njs

PX 719ᶜ 10-(10:27)


Digital
Computing Center
Numerical
Analysis
Mathematical
Analysis
Systems
and
Training
Operations
Programming

10-334

10:27

MATHEMATICAL SERVICES BRANCH

At the Air Force Armament Center the bulk of the machine computations are performed by two Directorates, the Directorate of Ballistics and the Directorate of Technical Support.

Within the Directorate of Technical Support the Mathematical Services Branch is responsible for the reduction of data on each test project to insure that data are suitable for analysis and/or inclusion in the final report. This Branch is divided into three Sections; Test Data, Computation and Equipment Maintenance.

The Test Data Section establishes mathematical and assessment procedures for processing test data; receives, stores, edits, and reviews raw and processed data; assists other agencies in programming problems; spot lchecks processed data, accomplishes assessment on selected portions of test data; prepares data for submission to project and analysis personnel.

The Computation Section accomplishes computation on the electronic digital computer; assists mathematicians in working out computation and reduction techniques and in selecting instrumentation methods compatible with computing and analyzing equipment; operates analog-to-digital converts, plotters, electronic analyzers, recorders, and computer output devices, develops techniques and procedures applicable thereto.

The Equipment Maintenance Section maintains data reduction and computing equipment of the Center; provides service assistance in the electronic computer and associated equipment; accomplishes minor modification of existing equipment as required by Project Mathematicians.

Each of these Sections are divided into operating groups with defined responsibilities in specialized fields as indicated on the organization chart.

Within the Directorate of Ballistics the Ballistics Computations Officer performs necessary data programming and computations utilizing existing facilities to greatest extent possible; provides consultant services to develop new data reduction and computation techniques.

A third organization, the Analysis Division, also initiates computing requirements at the Air Force Armament Center. Within this Division the Test Analysis Branch recommends priorities for all the Center's computing facilities and associated functions; applies numerical analysis techniques and performs coding and computations of analysis problems on the Center's computers as required. The Research Branch originates improved and/or new computing techniques and procedures where existing ones are inadequate or non-existent, except in the ballistics area; investigates, on a continuing basis, computing methods and techniques employed elsewhere, for possible use at the Center.

In addition the Applied Mathematics Division of the Directorate of Statistical Services, Air Proving Ground Command, assists in programming and operating Air Force Armament Center projects on the computers available in that Command.

PX 71900-10-(10:28)

UNPACKED FLOATING POINT CARD READ

7 400	00742	56	00000	30000	EXIT
7 401	00743	75	31777	72403	STORE
7 402	00744	11	00001	74001	ES
72403	00745	75	30352	00747	ROUTINE
72404	00746	11	72400	00742	TO ES
72405	00747	11	00742	20000	SET
72406	00750	36	01264	72432	REPEAT
72407	00751	35	01305	20000	SET
72410	00752	16	20000	00770	EXIT
72411	00753	55	00742	20025	TEST FOR
72412	00754	44	00757	00756	ES
72413	00755	00	00000	00114	ADDRESSES
72414	00756	32	00776	00000	ADDRESS MODIFICATION
72415	00757	55	20000	00017	
72416	00760	37	00760	00761	
72417	00761	16	20000	00763	SET ACQUISITION
72420	00762	11	00777	01024	ERASE INSTRUCTION
72421	00763	71	01264	30000	ACQUIRE CONTROL WORD
72422	00764	55	20000	00006	
72423	00765	37	00760	00754	X
72424	00766	55	20000	00017	
72425	00767	37	01001	01004	TP SIBRPITOME
72426	00770	75	31777	30000	RESTORE E S
72427	00771	11	74001	00001	AND EXIT
72430	00772	75	31777	72432	RESTORE E S
72431	00773	11	74001	00001	AND
72432	00774	30	00000	0000	REPEAT
72433	00775	02	00000	00104	
72434	00776	74	00000	00000	
72435	00777	11	00000	00000	
72436	01000	30	00000	0000*	REPEAT
72437	01001	45	00000	30000	EXIT

THE RAMO-WOOLDRIDGE CORPORATION
LOS ANGELES 45, CALIFORNIA

UTILITY ROUTINE LIBRARY

TABLE OF CONTENTS

PX 71900-10-(71)

MDP-0	12-09-55	THE FLEXOWRITER MEMORY DUMP, REVISED
MDP-1	12-09-55	THE BIOCTAL MEMORY DUMP, REVISED
MDP-2	12-09-55	THE OCTAL CARD DUMP
MDP-3	12-09-55	CHANGED WORD POST MORTEM
MDP-4	05-01-56	OCTAL CARD DUMP
MII-0	05-01-56	MANUAL INSPECTION AND INSERTION
MTI-0	11-30-55	LINEAR MATRIX EQUATION SOLVER
MTI-1	07-26-56	LINEAR MATRIX EQUATION SOLVER, FLOATING POINT
MTI-2	08-03-56	LINEAR MATRIX EQUATION SOLVER, COMPLEX
NRT-0	12-01-55	NTH ROOT ROUTINE
NUI-3	05-01-56	NUMERICAL INTEGRATION BY THE GILL METHOD
NUI-4	05-10-56	FLOATING POINT GILL METHOD
NUI-5	06-20-56	GILL METHOD, COMPLEX
POL-0	09-04-56	ALGEBRAIC EQUATION SOLVER
RAN-0	05-20-56	NORMALLY DISTRIBUTED PSEUDO RANDOM NUMBERS
RPH-0	05-23-56	COLUMN HEADING ROUTINE
SAM-0	08-09-55	AUTOMATIC SAMPLER
SIN-0	05-01-56	CENTRAL EXCHANGE SINE-COSINE ROUTINE
SIN-1	05-01-56	POLYNOMIAL MULTIPLY SINE-COSINE ROUTINE
SIN-2	05-01-56	SMALL ANGLE SINE-COSINE ROUTINE
SIN-3	08-10-55	FLOATING POINT SINE-COSINE ROUTINE
SIN-4	05-15-56	FLOATING POINT SINE-COSINE
SNI-1	06-15-56	ARCSINE-ARCOSINE ROUTINE
SNI-2	08-01-56	ARCSINE-ARCOSINE, FLOATING POINT
SQR-0	05-01-56	SQUARE ROOT ROUTINE
STT-0	12-09-55	STORAGE TO MAGNETIC TAPE TRANSFER
TNI-0	05-01-56	ARCTANGENT ROUTINE
TNI-1	06-15-56	FLOATING POINT ARCTANGENT ROUTINE
TST-0	12-09-55	MAGNETIC TAPE TO STORAGE TRANSFER
URT-1	WRITE UP NOT AVAILABLE	
URT-3	WRITE UP NOT AVAILABLE	

CUMULATIVE ERRATA 08-15-56

BULLETINS #12 AND #14

THE RAMO-WOOLDRIDGE CORPORATION
Los Angeles 45, California

RAMO-WOOLDRIDGE PROGRAMMING AND OPERATING CONVENTIONS

1. Drum addresses 55359 to 56383 (76000b to 77777b) have been reserved for the drum image of electrostatic storage. These 1024 cells will be used by those service routines which operate in electrostatic storage and which must restore electrostatic storage after performing their functions. This part of the drum should not be used by the programmer to store part of his production program (Note: Drum addresses start at 40000 in decimal notation).

2. Drum cells 52287 to 55358 (70000b to 75777b) are reserved for the storage of service routines and are not, in general, available for general program use.

3. Drum cells 48191 to 52286 (60000b to 67777b) will be used for the assembly data, assembly program, and subroutines. These cells are available for general program use, since the information will be stored in these cells only during program assembly. If more space is required for subroutines, cells 57777b and lower may be used.

4. Electrostatic stor ge cells 13 through 22 (15b - 26b) have been set aside for a ten-word constant pool as follows:

13	Zero	00015	00 00000 00000
14	Six Bit Extractor	00016	00 00000 00077
15	Advance u	00017	00 00001 00000
16	Advance v	00020	00 00000 00001
17	Advance u and v	00021	00 00001 00001
18	Decimal 72	00022	00 00000 00110
19	Alternator	00023	52 52525 25252
20	Operation Code Extractor	00024	77 00000 00000
21	"n" Extractor	00025	00 07777 00000
22	(not yet assigned)	00026	

5. Cells 23 through 32 (27b through 40b) have been set aside for temporary storage. These temporary cells should be used in the strictest sense of the word and the programmer should not assume information will remain in these cells unchanged in passage from one portion of a program to another.

6. There will be three methods of starting the computer depending upon the amount of information stored on the drum and magnetic tape as follows:

   a. If the Ferranti read-in program is stored on the drum in its normal place along with the other utility routines, a tape can be read in by transferring control to cell 40001.

PX 71900-10-(71)

b. If the service routines are not on the drum, an MT start is performed which reads the service routines onto the drum from magnetic tape Unit = 0. During this transfer, check sums are completed and checked against sums previously obtained upon storage of the routines on tape.

c. In the event the service library data has been removed from the magnetic tape Unit = 0 and the drum due to engineering maintenance or another reason, a bootstrap procedure will be used to read the service routine library and Ferranti read-in routine into appropriate positions in the computer from either paper tape or binary cards.

7. Certain jump instructions will be read into cells 40001 through 40040 along with the service library data. The jump instruction will cause control to revert to the routines as indicated in the list of 40000 group entrances.

The jump word in 40000 is placed there during read-in of the program tape prepared by the assembly routine. The remaining jump words will be assigned as necessary. The usage of these cells to transfer control to the appropriate service routines will result in the following advantages: the cells are easily remembered; the PAK is easily set to addresses in the indicated range by means of the MD start; and the cells serve as "symbolic entrances" since the position of the service routines on the drum can be changed without changing the entrance address, and the programmer need not concern himself with the knowledge of the new location.

8. In programming production problems it is desirable that conditional stop orders be included so that the computer operator can halt the progress of a production run when it becomes necessary. Such stops should be strategically placed in short loops and at a point where the contents of A and Q are not relevant. It will be necessary, therefore, that the programmer supply the computer operator with the following data:

a. How to set up the stop order(s). That is, for example, which MS selecting switches must be set to "stop".

b. What are the areas of ES and the drum used by the program.

THE RAMO-WOOLDRIDGE CORPORATION
Los Angeles 45, California

PROGRAMMING REMINDERS

The following list of Programming Reminders has been compiled using the experience of the members of the programming staff. The list has been divided into two groups: general, and those reminders pertaining to the One Pass Assembly Routine (RAWOOP, CMP-0).

General Reminders

1. An index junp changes the contents of the accumulator.

2. Use (n-1) with an index junp, assuming the index jump is performed after the operation.

3. Clear the accumulator before using an MA instruction.

4. A magnetic tape command changes the contents of the Q-register.

5. The order of the j and n in a repeat instruction is as follows:

   RP j n w

6. The B register occupies A_{70}, A_{69} ......A_{35} and is different from AL. Not only can one perform such operations as TP B A but the fault circuitry will not prevent one from jumping to B, as it will for the A register. The operation SF B k will result in the scale factoring af A + B. (This is applicable only to the Ramo-Wooldridge 1103.)

7. When the computer is halted in the middle of a repeat instruction sequence, the PAK register contains the complement of j(n-r).

8. When using an RP instruction followed by a TJ or EJ instruction, the rightmost fifteen bits of the Q-register contain j(n-r) upon jumping out of the sequence. Since j is present n-r must be obtained by an extraction rather than with a TV instruction.

9. It is interesting to note that much can be said concerning error growth for unrounded multiplication and division. The following rules hold:

   Multiplication (axb)

   the truncated product (contents of B register) is too small if $axb > 0$

   the truncated product is too large if $axb < 0$

   the absolute value of the product is always too small

Division (a/b)

the actual quotient is too small if $b > 0$

too large if $b < 0$

Hence, one can predict, for example, that $\sum_i a_i/b_i$ will be too small if $b_i > 0$. This bias precludes the possibility of employing the usual formulas for probable error which assume a normal distribution of error about a mean zero.

Reminders for the One Pass Assembly Routine

1. If it is desirable to refer to the constant and temporary storage pools with symbolic addresses, directory cards should be included for them.

2. Rawoop converts decimal numbers to binary numbers occupying 35 bits plus a sign bit.

3. When converting drum addresses, 40000 in decimal is equivalent to 40000 in octal.

4. When specifying the beginning of a region, one must use the first symbolic address of a block. For example, D 03M00 00500 must be used and not D 03M15 00515.

5. Since SNAP addresses are limited to 9 bits the A, Q and B registers cannot be addressed directly.

THE RAMO-WOOLDRIDGE CORPORATION
Los Angeles 45, California

The Utility Routine Library Handling Package for Paper Tape Input

Normal Operation

During normal operation, the Service Routine Library is stored on the drum. In order to use one of the routines, control is transferred to one of the low-numbered drum addresses in the 40000b channel (see the list of "Service Routine Starting Addresses").

Details concerning the operation of these routines and their locations can be found in the write-ups.

MT Start

If, at any time, the library stored on MD is destroyed by a program, or because the drum interlace has been changed, or for some other reason, the entire library may be loaded onto MD from magnetic tape. Selecting MT Start and starting effects loading of the service routine library from MT zero. PAK is set to the FRI-0 starting address upon completion of the transfer. Selecting MT Start, setting $A_R = 1$, and starting effects loading of the service routine library and the assembly program and subroutines from MT zero. PAK is set to the CMP-0 starting address upon completion of the transfer.

Bootstrap

Since the Ferranti reader requires a programmed read in, it is necessary to "bootstrap" into the machine when no input routine is stored in memory. The procedure devised to load an input program involves the use of one binary card (since this method requires the fewest number of instructions to be loaded manually). It is necessary to key in manually only four words which perform the read in of one binary card (24 words) and transfer control to these 24 words.

This binary card contains a simplified Ferranti Input Routine (FRI-0) which then begins to read in the service routine library paper tape. This tape contains at its beginning the regular FRI-0 input routine and instructions transferring it to its proper location on MD. When FRI-0 has been loaded on MD control is transferred to it and FRI-0 reads in the remainder of the tape.

Following FRI-0 on the tape are the library, a Magnetic Tape to MD transfer routine (URT-1), and an MD to Magnetic Tape transfer routine (URT-3). When the complete paper tape has been read in the computer halts with the library loaded on MD. PAK is set to 40000 by an MD Start, the machine is started and URT-3 transfers the library and URT-1 to magnetic tape unit zero. The computer then halts with PAK set to 40001, the FRI-0 starting address.

Detailed Descriptions of Routines

Detailed descriptions, operating instructions, and codes for the routine mentioned above are included in the notebook.

PX 71900-10-(71)

Operating Instructions for the Bootstrap Procedure (Loads MD and MT zero with Service Routine Library)

1. Put the binary card deck in reader making sure that card reader is set for two fields and that all three switches on the reproducer are away from the card hoppers. Also be sure that there are cards in the punch hopper as the reader will not operate without them.

2. Put the library paper tape in the Ferranti Reader.

3. Position MT zero at the dead space immediately preceeding the first block. (Maintenance people will perform this function if requested).

4. Key in the following program:

00104	17 00000 00104	pick card
00105	17 00000 00105	read and pick card
00106	75 30030 00000	read one binary card
00107	76 10000 00000	then jump to 00000

5. START at 00104. The computer reads in one card containing the simplified Ferranti routine, then switches control to this routine which reads in FRI-0 and the necessary orders to transfer it to MD. After FRI-0 has been placed on MD, control is transferred to it and it reads in the library tape and the MT to MD and MT transfer routines. The computer then halts with the MS instruction 56 00000 40001.

6. Select MD Start

7. Set low order octal digit of Q to the address of the MT unit desired.

   To load tape zero, omit this step of the procedure.

8. START. The MD to MT routine loads the MT to MD routine, 40001b thru 40040b, 70000b thru 75777b, and 60000 thru 67777 onto MT zero. The computer then halts with the MS instruction 56 00000 40001, setting PAK to the FRI-0 starting address.

THE RAMO-WOOLDRIDGE CORPORATION
Los Angeles 45, California

Pool of Flexowriter Codes

Specifications

Identification Tag:	FLX-O
Type:	Constant Pool
Storage:	17 cells, addresses 75757b thru 75777b
Note:	This pool of certain flexowriter codes has been established in order to prevent duplication of storage in the Service Routine Library. See the listing for details.

Approved by: W. F. Bauer June, 1956

```
D           FLX00 55343                        75757  00 00000 00000
FLX00             37 B        F 0              75757  00 00000 00037
FLX01             52 B        L 1              75760  00 00000 00052
FLX02             74 B        E 2              75761  00 00000 00074
FLX03             70 B        X 3              75762  00 00000 00070
FLX04             64 B        O 4              75763  00 00000 00064
FLX05             62 B        W 5              75764  00 00000 00062
FLX06             66 B        R 6              75765  00 00000 00066
FLX07             72 B        I 7              75766  00 00000 00072
FLX08             60 B        T 8              75767  00 00000 00060
FLX09             33 B        E 9              75770  00 00000 00033
FLX10             45 B        R CAR RETURN     75771  00 00000 00045
FLX11             04 B          SPACE          75772  00 00000 00004
FLX12             57 B        C SHIFT DOWN     75773  00 00000 00057
FLX13             47 B        O SHIFT UP       75774  00 00000 00047
FLX14             51 B        D TAB            75775  00 00000 00051
FLX15             42 B        E PERIOD         75776  00 00000 00042
FLX16             56 B        S MINUS          75777  00 00000 00056
START
```

THE RAMO-WOOLDRIDGE CORPORATION
Los Angeles 45, California

CUMULATIVE ERRATA

CVF-0 11/14/55

Page 5, line 12 from the bottom should read "n = (50000b)" and not "n + 1 = (50000b)".

DET-2 7/25/56

Page 4, Line 00R10 should read "SP A0000 00016 00776 31 20000 00020"

EGN-0 5/1/56

Page 8, line 14 should start "or the ERA paper tape reader)....."

Page 16, line 7 should start "if $n \leq 38$,.....

EXT-3 8/10/55

Page 1, drum assignment should read "63766 b through 64044 b."

HTO-0 7/25/55

Page 1, drum assignment should read "62504 b through 63037 b".

MDP-3 12/9/55

Page 2, line 12, reference to MDP-2 should read MDP-4.

Page 2, line 14, should start "each card contains six words".

NRT-0 12/1/55

Page 2, line 5 should read "$2 \leq n \leq 2^{12}$"

NUI-3 5/1/56
NUI-4 5/10/56
NUI-5 6/20/56

In each of these Gill Method routines it should be noted under "Operating instructions" that the cells reserved for the q_1 must be set to zero by the programmer initially and whenever a discontinuity occurs.

It has been pointed out that the error analysis in Gill's paper is not applicable to these routines.

NUI-5 6/20/56

Page 1 "type" should read "subroutine, available on cards for assembly". "Assembly Routine Spec" line should be replaced by "Regions used: GIL, GIN, GIM, GCN"

SIN-4 5/15/56

Page 1, "Storage" should read "65 words total program storage. 5 words temporary storage pool used, addresses 27 b through 33b".

PX 71900-10-(71)

THE RAMO-WOOLDRIDGE CORPORATION
Los Angeles 45, California

Utility Routine Transfer Drum to Magentic Tape

Specifications

Identification Tag:	URT-3
Type:	Service routine (part of library loading package)
Storage:	110 instructions, addresses 00045b thru 00222b
	8 constants in program, addresses 00223b thru 00232b
	All of ES is used for temporary storage but not included with the program
	118 words total program storage, addresses 00045b thru 00232b
	The temporary and constant storage pools are not used by this routine
Entrance:	40036b
Machine Time:	100 seconds approximately
Mode of Operation:	Fixed point

Coded and Checked by:	R. Beach	August, 1955
Approved by:	W. F. Bauer	August, 1955

Description

Upon being entered the routine first bootstraps itself and URT-1 into ES, then sets up all references to magnetic tape to correspond to the unit selected when URT-3 is activated. URT-1 is transferred to ES beginning at cell 1700b.

The contents of cells 40001b thru 40040b and 70000b thru 75777b are then summed and the sum placed in 01774b and 01775b. The contents of cells 01700b thru 01775b are then summed and the sum placed in 01776b and 01777b. The information in cells 01700b, 70000b thru 75777b, and 40001b thru 40040b are then transferred to MT in that order.

The contents of cells 60002b thru 67777b are summed and the sum placed in 60000b and 60001b.

STT-0 is entered to dump the information in cells 60000b thru 67777b (the subroutine library consisting of RAWOOP and the subroutines).

URT-3 computes the sum of all information placed on MT, rewinds MT to its original position and reads back the data from MT, summing as it reads.

If the sum is correct, a BM instruction is given to return MT to its original position and computation halts with PAK set to 40001b, the FRI-0 starting address.

Operating Instructions

1. Select MD Start.

2. Set the number of the MT to be loaded in the low order octal digit of Q.

3. Start.

   URT-3 transfers the complete library to MT and halts with the MS instruction 56 00000 40001 after a successful transfer.

Alarm Conditions

If the sum of data read back from MT is not correct the alarm routine is entered; the tag word URT-3 and the address 00070 are printed on the flexowriter. The sum of the data on MT appears in A.

Restarting at this time initiates another transfer to data.

Warning

1. It is advisable to position MT at the first block before loading so that the MT can be repositioned, manually if necessary.

2. After a successful transfer the machine halts but MT is still rewinding to its original position. If a master clear is executed and the machine started a reference to the rewinding MT (before the rewinding is complete) will cause trouble. If no master clear has been executed the machine will wait for the reversing to be completed.

PX 71900-10-(71)

THE RAMO-WOOLDRIDGE CORPORATION
Los Angeles 45, California

Utility Routine Transfer-Magnetic Tape to Drum

Specifications

Identification Tag:	URT-1
Type:	Service routine (part of Library Loading Package)
Storage:	57 instructions, addresses 00000b thru 00070b
	1 constant in program, address 00073b (remaining constants stored with instructions)
	The remainder of ES is used as temporary storage
	The constant and temporary storage pools are not used by this routine
Entrance:	MT Start
Machine Time:	Approximately 15 seconds for successful transfer of the service routine library only, or approximately 35 seconds for transfer if CMP-0 and the subroutine library are included.

Coded and Checked By:	R. Beach	April, 1955
Approved by:	W. F. Bauer	August, 1955

Description

This routine is located in the first two blocks of the magnetic tape unit used for the library (normally MT 40) and is specifically designed to transfer the library from magnetic tape to magnetic drum.

It operates in three different modes, the mode of operation having been selected when it was activated. Mode No. 1 loads addresses 40001b thru 40040b and 70000b thru 75777b only. Mode No. 2 loads these addresses and addresses 60000b thru 67777b. Mode No. 3 advances past the library and reads in the first block of the ERA maintenance routine loader, then transfers control to that loader.

This routine does not save the contents of ES since it is assumed that it will be used only when a complete reloading of the computer memory is necessary. An MT Start reads in the first 32 words of the routine and starts operation. The routine first reads in an additional 32 words from MT (remainder of the routine itself) and then checks its sum, which is stored at the end of the second block. In doing this, it also checks the sum of the service routine library which is stored in the second block.

After a successful sum check the routine reads in the 96 blocks needed to fill 70000b thru 75777b. Twenty-four blocks are read in at one time and transferred to MD, then read back into ES and summed. When all 96 blocks have been transferred the routine reads in one more block and transfers this into 40001 thru 40041, reads it back into ES, sums, and adds the sum to the sum of the 96 blocks previously transferred. This computed sum is then checked against the correct sum.

If Mode No. 1 has been selected a rewind instruction is given and the computer halts with the MS instruction 56 00000 40001, setting PAK to the FRI-0 starting address.

If Mode No. 2 has been selected, TST-0 is activated to read in RAWOOP and the subroutine library. A rewind instruction is given and the computer then halts with the MS instruction 56 00000 40010, setting PAK to the CMP-0 starting address.

Operating Instructions

I. To transfer the service routine library only

1. Select MT Start

2. Change PCR (if necessary) to select the proper MT unit

3. Start. The routine loads 40001b thru 40041b and 70000b thru 75777b and halts with the MS instruction 56 00000 40001, setting PAK to the FRI-0 starting address. Successful transfer takes about 15 seconds.

II. To transfer the service routine library, CMP-0, and the subroutine library

1. Select MT Start

2. Change PCR (if necessary) to select the proper MT unit.

PX 71900-10-(71)

3. Make A = 1

4. Start. The routine loads 40001b thru 40040b, 70000b thru 75777b, and 60000b thru 67777b and halts with the MS instruction 56 00000 40010 setting PAK to the starting address of CMP-0. Successful transfer takes about 35 seconds.

III. To load ERA Maintenance Routines

1. Select MT Start.

2. Change PCR (if necessary) to select the proper MT unit.

3. Make A $\geq$ 2

4. Start. The routine advances the MT unit and loads the 229th block into ES placing the first word in cell zero. The number in the accumulator is reduced by 2 and left in A for the ERA loader to use in selecting its mode of operation. Control is then transferred to cell zero.

Alarm Conditions

1. If the machine halts on a final stop almost immediately after an MT start the transfer routine is not in ES correctly.

   Select MT start and start for another transfer. If the second transfer is not successful revert to the bootstrap procedure to load the library.

2. If the flexowriter prints an "e" and the machine halts on a final stop the sum of the library transferred to the drum is not correct.

   Select MT start and start for another transfer. (Wait for rewinding to be completed before executing start).

3. When operating in Mode No. 2 TST-0 is activated after address 40001b thru 40040b and 70000b thru 75777b have been loaded successfully. If the sum test fails while loading addresses 60000b thru 67777b, the alarm routine prints the tag word TST-0 and the address 75777b. Starting causes rewind and another MT transfer to addresses 60000b thru 67777b.

Warning

After a transfer the computer halts but MT is still rewinding to its original position. If a Master Clear is executed and the machine started, a reference to the rewinding MT (before rewinding is complete) will cause trouble. If no Master Clear has been executed the machine will wait for the rewinding to be completed.

THE RAMO-WOOLDRIDGE CORPORATION
Los Angeles 45, California

THE RAMO-WOOLDRIDGE ONE-PASS ASSEMBLY ROUTINE

Specifications

Identification Tag:	CMP-0	
Type:	Service Routine	
Entrances:	40000b and 40010b	
Coded by:	Jules Mersel	
Code Revised by:	Thomas Tack	October 1955
Approved by:	Walter F. Bauer	October 1955

PX 71900-10-(72)

Description

The Ramo-Wooldridge one-pass assembly program (RAWOOP) is designed to translate an 1103 program originally coded in symbolic, regional, octal, and decimal form into its final octal form.

The program will accept instructions with symbolic addresses and numerical data in binary or decimal form. It will cause subroutines to be appropriately assembled into the program. The result of assembling a program will be output in a form which facilitates program check out and rapid read in of the translated data.

Input-Output

Punched cards are used as input for RAWOOP. The punched card contains one 1103 word with remarks or contains an instruction to the assembly program with remarks. RAWOOP's output is via both punched cards and punched paper tape. The output card contains an exact duplicate of the corresponding input card plus an octal translation of the input data. The programmer can obtain a side by side listing of his untranslated program, remarks, and translated program by listing the output deck on associated equipment such as the IBM 407.

The punched paper tape is a seventh-level bioctal tape complete with insert and check addresses, and is used to read the translated program into the 1103 by either the ERA photo-electric reader or by the Ferranti tape reader with an appropriate read in program. A leader and trailer is automatically included on each tape prepared.

Input and Output Cards

The input and output cards are standard 80-column, 12-row cards. The allocation of the information with respect to the card columns is as follows:

1-5	symbolic address of the untranslated word
7-10	1103 operation characters or pseudo-instruction symbols of the untranslated word
12-16	u address of the untranslated word
18-22	v address of the untranslated word
24-26	decimal scaling information for the untranslated word
28-30	binary scaling information for the untranslated word
32-43	alpha-numeric remarks

It is noted that symbolic coding for SNAP (Interpretive floating point package) deviates from these conventions (see pg. 6). In addition to the above columns, the output cards contain the translated information in columns 47-67.

<u>On the input cards, zeros need not be punched.</u>

FIGURE 1

THE RAMO-WOOLDRIDGE CORPORATION

PAGE______ OF______

PROBLEM______

Line	ADDRESS	OP	U	V	± D	± B	COMMENTS
1	D		1C	512			THIS IS NOT
2	D		99Z	630			
3	D		12T	49010			A PROGRAM.
4	D		73L	183			
5	D		T	2			
6	1C	RP	3 19	1C 2			IT IS A LIST
7	1C 1	TN	12T 5	Q			
8	1C 2	SP	99Z13	9			OF EXAMPLES
9	1C 3	MJ	3	99Z 1			
10	1C 4	LA	12T 4	2 18			OF THE TYPES
11	1C 5	ZJ	99Z	99Z02			
12	1C 6	-1	23456	78901	4	18	OF WORDS
13	1C 7	3	123		- 6	22	
14	1C 8	1	6793		8	- 10	ACCEPTABLE
15	1C 9	TP	Q	A			
16	1C10	21	12T28	12T12	BRR		TO CMP-O.
17	1C11	RJ	73L 1	73L 2			
18	1C12	SJ	99Z23	99Z30			
19	1C13	1	024		3	F	
20	1C14		77777		B		
21	73L	SUB	51018	23 4			
22	START		01C00				
23	DIC00	72	03421	01C29	BBR		
24	DIC01	17	D1C29	03421	BRB		
25	DIC02	27	1RC29	DBC28	BRR		
26	DIC03	ADSU	K9P00	K9P01			
27	DIC04	MPDV	K9P00	K9P01	B		
28	DIC05	PMLD	K9P00	K9P01		S	
29	DIC06	STF1	K9P00	K9P01	BS		
30	DIC07	FLRT	K9P00	K9P01	B	S	
31	DIC08	RDPD	00000	00010			

CODED BY______ CHECKED BY______

FORM 64 SOCAL STAT.

SYMBOLIC — LOCATION, OPN. (← FOR DATA), U, V, ± D, ± B

REMARKS

ABSOLUTE — LOCATION, OPN, U, V

THE RAMO-WOOLDRIDGE ONE PASS ASSEMBLY PROGRAM

PX 71900-10-(72)

Speed of Assembly

Due to the 1103's ability to read and punch cards while simultaneously punching paper tape, RAWOOP takes only a few seconds more than the total card reading time to execute its entire translation. Errors do not necessitate complete re-assembly and consequently the routine is exceedingly economical in its use of machine time (see below).

Symbolic Addresses

A five character form, in keeping with 1103 machine form, is used for symbolic addresses. The first three characters designate the region of the address while the last two characters are the sequence number of the address within its region.

For example, DRJ00 is the zeroth address in region DRJ and 01C19 is the nineteenth address in region 01C (01C00 is the zeroth address in region 01C). In keeping with the one-pass nature of RAWOOP, the sequence numbers are consecutive decimal numbers. The absolute address assigned to 01C19 is nineteen greater than the address assigned to 01C00. Thus, the address structure has a regional character.

As indicated above, the first three characters of the relative address or region are alpha-numeric.

The absolute address for region 000 (all zeros) has already been chosen in RAWOOP to be zero. Consequently, 00029 would have 00035 as its octal translation; absolute machine addresses up to 99 will be correctly translated. Furthermore, the regions A00, B00, and Q00 are best avoided, as in some cases reference to them will be interpreted as addressing the A, B, or Q registers respectively.

The assembly program recognizes the alphabetic letter "O" as different from the numeric zero but in order to avoid confusion the programmer will probably not want to use symbolic addresses involving the letter "O".

A, B, and Q Addresses

The accumulator, the B register (accumulator bits A_{70} - A_{35}), and the Q register must be addressed by putting an A, B, or Q, respectively, in the leftmost column of either the u or v fields. The remaining four columns of the field may have no punches or zero punches. The octal translations of A, B, or Q are 20000, 30000, and 10000 respectively.

Addressing Involving j, k, and n

The command structure of the 1103 is such that the u and v addresses at times contain numbers rather than machine addresses (as in the case in the SPuk, RPjn, w and MJjv commands). The representation of j, n type instructions uses j as the first character and n as the last four characters with the quantities j and n written as decimal numbers. Thus the j, n number 30199 is translated to 30307b. No distinction is made between j, n addresses and j addresses. If the programmer desires to use the last four characters of a j, type address to store a number (this cannot be used to store a relative address) he may do so accounting for the fact that these four digits will be treated as the n portion of a j, n type address.

In the 1103 the k address structure is used to represent left circular shifts from 0 to 127 places. However, since the internal hardware occasionally makes it desirable to have the first octal bit of a k address be a number other than zero, k addresses will be treated in exactly the same manner as j, n addresses; for example, 20017 becomes 20021b.

Octal-Symbolic Words

In order to allow the programmer to mix octal and symbolic addresses in the same instruction three special types of words ("BBR","BRB", and BRR") have been included in RAWOOP. The first letter of the triad refers to the operation, the second, to the u address and the third, to the v address. R means symbolic (regional) and B means octal (binary). Thus, a BBR word has its operation and u address octal and its v address regional. The flag BBR goes into the D card field (columns 24-26). These flags cannot be attached to SNAP commands. Examples can be found on lines 23-25 of figure 1.

Void Addresses

Certain of the 1103 commands such as FS--, RJj,n-- have ignored addresses associated with them. All such addresses are treated by RAWOOP as if they were relative addresses and are available for the storage of pre-setting addresses. An all-zero address, of course, is translated into all zeros.

Directory Cards

In a one-pass assembly program, it is necessary that at the beginning of the program sufficient information is supplied to enable all symbolic addresses to be assigned absolute addresses. RAWOOP does this by means of directory cards. A directory card has a D punched in column 1, the base word of the region (e.g. 01C00) in the u address columns, and the absolute decimal address of the base word in the v address columns. For examples, see figure 1. RAWOOP can handle up to 73 directory cards in any one assembly. The programmer is cautioned not to follow his directory cards with program cards destined for a region D00. Such an ordering would prevent RAWOOP from differentiating between the two types of cards.

For purposes of assigning decimal addresses to the drum, the convention was adopted that octal address 40,000 on the drum has the decimal address 40,000. Thus the drum addresses range decimally from 40,000 - 56,383.

Symbolic Addresses of Program Data

With the exception of the D cards and the START card, all the input cards have a relative address punched in columns 1-5. This address is the address of the word to the right and, in conjunction with the decimal address of the corresponding directory card, completely determines the memory location into which this card will be read by the bioctal tape punched by RAWOOP.

Commands

The 1103 alphabetic representation of the commands is used. These two letter combinations, such as RA for the "replace add" or "21 command", are entered into columns 9 and 10. All the standard 1103 commands are recognized by RAWOOP and this recognition implies knowing whether the addresses associated with the command are of the u, v, the jn, v, or of the u, k types.

In addition to the standard 1103 commands, the special commands IP, PM, and MM are recognized. The PM and MM commands and the availability of the B register for addressing are modifications on the Ramo-Wooldridge 1103. The PM command is a "polynomial multiply" command for polynomial evaluation whose octal equivalent is 24; the MM command is the "modified multiply-add" (faster in operation than MA) whose octal equivalent is 25. None of these modifications are used in the operation of RAWOOP; the program will operate on any 1103 with reproducer and high-speed punch.

The IP commands are treated as if the command structure were IPuv. However, for use in SNAP, a pseudo command is available. This has the form ABCDuv, where AB and CD are alphabetic SNAP commands, while u and v are symbolic addresses restricted in their assignment to the lower half of electrostatic storage. This form is translated by RAWOOP into 14A'u'C'v' where A' and C' are the octal representations of the SNAP commands AB and CD respectively, while u' and v' are 9 bit numbers. A' and u' are packed into the u address of the translated command and occupy the left 6 bits and right 9 bits respectively. C' and v' are similarly placed in the v address. On the input card AB occupies columns 7 and 8, while CD appears in columns 9 and 10.

SNAP provides for a B-box whose contents can be added to the address of a SNAP command before its execution. To cause RAWOOP to translate SNAP commands so that this signal will be given, a "B" is placed in column 25 of the D field to modify the u address, or in column 29 of the B field to modify the v address of the SNAP command. Upon finding a "B" in column 25 and/or 29, RAWOOP adds two (2) to the corresponding SNAP operation.

If the programmer places the letter "S" in column 26 of the D field and/or column 30 of the B field of a SNAP pseudo command, RAWOOP adds one (1) to the corresponding SNAP operation. On execution of this command, the contents of the F register will be stored in the address associated with this command.

For example, if symbolic address K9P00 has its octal address equal to 00500, the translation of lines 26-31, of figure 1, are respectively:

14	04500	10501
14	16500	20501
14	24500	31501
14	37500	40501
14	46500	51501
14	70000	74010

It is sometimes desirable to place a relative address in a word that has zeros for its command code. To allow for this, the command 00 (zeros) is recognized and translated into 00. The u and v addresses must be symbolic addresses. As usual, however, an address of five zeros is translated into five zeros.

Decimal Numbers

Decimal numbers are presented to RAWOOP as normalized numbers times a power of ten. The programmer also states the binary scaling factor to be applied to the resulting rounded binary number. For example, -739.1 is presented as -7.391×10^2.

All decimal numbers are normalized so that their absolute value lies between 1 and 9.99999 99999.

The sign of the number is in column 9, the integral part in column 10, and the fractional part in columns 12-16 and 18-22. The power of 10 allowed is from -10 to +10. This exponent goes into columns 24-26. The desired binary scale factor goes into columns 28-30. For examples, in line 13, figure 1, the number to be translated is 3.123×10^{-6} with a scale factor of 2^{22}.

In all cases a minus sign represents a negative number and a zero or no punch a positive number.

Floating decimal numbers are presented to RAWOOP in the same manner as decimal numbers. However, instead of a binary scale factor being placed in columns 28-30, an F is punched in column 28. The converted floating point decimal number is in the form used by SNAP. That is, the left most bit is a sign bit, the next eight bits represent the exponent (increased by 128) and the twenty-seven bits on the right comprise the mantissa. In this form a negative number appears as the complement of the corresponding positive number.

Octal Constants

Octal constants can be inserted into the program using RAWOOP. The octal constant is entered into columns 9-10, 12-16, and 18-22. A "B" is punched into column 24 to signify that the number is in octal (binary) form. This feature implies that programs coded in the usual machine language will be correctly assembled.

PX 71900-10-(72)

Start Card

The signal to RAWOOP that all the cards have been received is a card with START punched in columns 1-5, and a relative address punched in columns 12-16.

The signal causes RAWOOP to insert into the translated program (in locations 00000b and 40000b) a manual jump to the octal translation of the relative address. For this reason, it is not possible to program anything in cells 00000b and 40000b.

After the resulting punched paper tape is read into the 1103, a magnetic drum start will start the problem at the relative address indicated in the u columns of the untranslated word.

Subroutines

RAWOOP is designed to translate subroutines coded in octal relative to 01000b (or 02000b) and stored on the drum, and include them into the main program. This is accomplished by one pseudo command, the "SUB" command.

Columns 1-5 of the SUB command contain the relative address at which the subroutine is to be stored. Columns 7, 8, and 9 contain the letters SUB.

Columns 12-16 contain the drum address of the cell in which the first word of the subroutine is stored for use by RAWOOP (this address is a "decimal drum address").

Columns 18, 19, and 20 contain the number of words (in decimal) in the subroutine. Columns 21 and 22 contain the number of constants (in decimal) which are placed at the end of the subroutine and are not to be modified during translation. In practice the information in these columns is simply copied by the programmer from a subroutine specification given on the title page of the subroutine write-up. Columns 25, 26, 28, 29, and 30 contain the relative address at which the subroutine is to be executed if this differs from the storage address. If these columns are blank or zero the storage address will be used as the execution address. The subroutine will be translated relative to the execution address but loaded (via the bioctal output tape) at the storage address.

The programmer is cautioned not to place a SUB card immediately after the directory cards.

The programmer has the options of including in the output cards the translated subroutine cards. With jump switch No. 3 off, the punching of these cards is suppressed. On the other hand, with this switch on, prior to starting the assembly, these cards are provided.

Subroutine Compiling and Constant Pools

There are two versions of RAWOOP (only four words differ). The non Ramo-Wooldridge version of RAWOOP assumes that subroutines stored on the drum are coded relative to address 01000b. The Ramo-Wooldridge version of RAWOOP assumes that subroutines are coded relative to address 02000b.

The Ramo-Wooldridge version of RAWOOP punches out on the bioctal tape a constant pool which is used in Ramo-Wooldridge subroutines. The pool is ten words long and starts in memory location 00015b. The sum of this pool is included in the memory sum recorded on the output tape. These constants are punched by RAWOOP before D cards are read and, of course, can be written over after they enter the high-speed storage. If constant pool cards are also desired they may be obtained by putting MJ 1 on before beginning the assembly.

Memory Sums

RAWOOP calculates the memory sum of all translated words including the 00000b and 40000b jump instructions. This sum is the double precision sum of the split extension of the translated words. The sum is both printed on the listing and read by the output paper tape into addresses 75202b and 75203b of the drum. The high order value of the sum is in 75202b. This placing of the sum is consistent with the Ramo-Wooldridge Ferranti read in routine. RAWOOP checks its own memory sum at the beginning of every assembly. Upon a memory sum disagreement, the routine prints "CMP-0".

Ordering of Input Cards

RAWOOP places the following requirements on the ordering of the input cards:

1. All directory cards must come first. The directory cards however, can have any order within themselves.

2. The START card must come last.

The cards actually giving the words of the program, subject to the restrictions already made, follow the directory cards. In the interest of minimizing the number of insert and check addresses on the output bioctal tape the cards should be in order within their regions. However, a correct output tape will be produced regardless of the order of the incoming program cards.

Suppression of Punched Paper Tape

If for any reason an input paper tape is not desired, the high-speed punching can be suppressed by turning on the j = 2 manually selective jump.

Storage Used by RAWOOP

RAWOOP uses all of high speed storage, cells 40000b and 40010b and stores itself in 60000b through 62117b. If subroutines are compiled, it also uses 43000b through 47777b.

PX 71900-10-(72)

Operating Instructions

The following points are important in the operation of RAWOOP:

1. The reproducer must be set for three fields.

2. The input cards must be placed face down with at least four blank cards following the START card at the back of the deck.

3. Both the typewriter and the high-speed punch must be turned on.

4. When RAWOOP is first read into the 1103, an MD start will store RAWOOP on the drum and start RAWOOP. All subsequent starts are made from address 40010b. When a run is completed, it stops with a manually selective stop to 40010. Pressing the START button causes another program to be assembled. The program checks its own memory sum at the beginning of an assembly and prints out "CMP-O" when the check sum fails.

5. All cards should be cleared from the read and write channels of the reproducer at the end of the program.

6. All reproducer switches must be on normal setting (away from the card hoppers on the Ramo-Wooldridge 1103).

7. All MS and MJ switches should be turned off except as noted above.

Error Detection

If more than 73 directory cards are entered the flexowriter prints "too many d cards" and the computer halts. It is not possible to continue the assembly after this error has occurred.

All other errors, including a D card occurring in the main deck will not stop the machine. For each card with an error on it, "ERROR" will be printed on the Flexowriter. The contents of the erroneous card will be ignored and will not effect the memory sum. The corresponding output cards will have "error" punched on them in the place ordinarily used for the translated information. If the error appears with respect to the START card, addresses 00000 and 40000 will be loaded with the word 45 00000 00000b by the paper tape.

The detected errors can be corrected by a second assembly of only the new corrected input cards, their associated D cards, and the START card. This will give a secondary input tape with its own memory sum.

Programming Reminders

1. If it is desirable to refer to the constant and temporary storage pools with symbolic addresses, directory cards should be included for them.

2. RAWOOP converts decimal numbers to binary numbers occupying 35 bits plus a sign. Decimal numbers not in the correct range will be translated incorrectly or can cause an error card to be punched.

3. When converting drum addresses, 40000 in decimal is equivalent to 40000 in octal.

4. When specifying the beginning of a region, one must use the first symbolic address of a block. For example, D 03M00 00500 must be used, and not D 03M15 00515.

5. To assemble a number of programs consecutively, it is necessary to have two cards between programs.

6. Directory cards must not be followed immediately by

   a. SUB cards
   b. program cards directed to region D00.

7. Attention is called to the use of A00, B00 and Q00 as regions. If these regions are addressed, reference may be made to the A, B, or Q registers.

8. Since the last two words of the translated program are start addresses inserted in 00000 and 40000, it is impossible to program for these cells using RAWOOP.

9. Correcting or replacing words on an assembled program tape will in most cases invalidate the sum checks at the end of the tape. The programmer has the following options for correcting or altering a program:

   a. re-assemble
   b. add the corrected words with proper insert and check addresses after the sum check
   c. re-assemble the incorrect cards only and enter into the machine the old and new assemblies in that order.

PX 71900-10-(72)

THE RAMO-WOOLDRIDGE CORPORATION
Los Angeles 45, California

Definite Integral Evaluation Routine

Specifications

Identification Tag:	DIE-0
Type:	Subroutine
Assembly Routine Spec:	SUB 49810 05804
Storage:	54 instructions, addresses 00P00 thru 00P07 00S00 thru 00S26 00N00 thru 00N18 4 constants in program, addresses 01C00 thru 01C03 58 words total program storage, addresses 00P00 thru 00P07 00S00 thru 00S26 00N00 thru 00N18 01C00 thru 01C03 10 words temporary storage pool used, addresses 00027b (00T00) thru 00040b (00T09) The constant pool is used by this routine
Drum Assignment:	Addresses 63122b thru 63213b
Program Entrance:	Address 00P02
Program Exit:	Address 00P01
Mode of Operation:	Fixed point

Coded by:	F. Meek	June 7, 1955
Code Checked by:	R. Bigelow	June 10, 1955
Machine Checked by:	F. Meek	July 7, 1955
Approved by:	W. Bauer	July 26, 1955

Description

Assuming that $y = f(x)$ is a tabulated function with equal increments in the argument $(x_0, x_1, x_2, \ldots, x_n)$, this routine will approximate the definite integral

$$\frac{1}{(x_n - x_0)} \int_{x_0}^{x_n} y dx$$

using either Simpson's rule or a modification of Simpson's rule.

The function values may be stored in ascending order of the argument in any block of consecutive storage cells.

At the time of entry into the subroutine the programmer must supply the value of n (that is, the number of intervals, one less than the number of points) and the address of the cell containing the first function value $y_0 = f(x_0)$.

The routine gives as a result an approximation to

$$\frac{1}{(x_n - x_0)} \int_{x_0}^{x_n} y dx$$

and the programmer must then multiply this result by $(x_n - x_0)$ to obtain the approximation to the integral itself.

Notation

$$I = \int_{x_0}^{x_n} f(x) \; dx, \quad J = \frac{1}{(x_n - x_0)} \; I.$$

I^* and J^* are approximations to I and J, respectively.

$\varepsilon = I^* - I.$

Range of y_i, J^*, and n

The only restriction on the y_i's is that they must be single precision fixed point numbers. The number of intervals n must be greater than one but can be arbitrarily large. The result J^* will be given scaled by the same amount that the y_i's were scaled.

In order to obtain the maximum significance for J^* the y_i's should be scaled as far to the left as possible.

Programming Instructions

Before entering the routine the function values y_i must be stored in ascending order of the arguments in consecutive storage cells.

PX 71900-10-(89)

1. Place $n \cdot 2^0$ in A.

2. Place the address of $y_0 = f(x_0)$ in Q_v

3. Enter the routine with the instruction RJ 00K01 00K02 (assuming 00K00 is the region that was assigned to the routine during assembly)

At the time of exit from the routine the result J* is left in A, scaled by the same amount that the y_i's were scaled. The numbers y_i have been left in their original state.

Mathematical Analysis

Let the equal increment of x be denoted by

$$h = x_i - x_{i-1} = \frac{x_n - x_0}{n} > 0, \quad i = 1, 2, \ldots, n.$$

Suppose I is to be approximated by a quadrature formula of the form

$$I^* = h \sum_{i=0}^{n} c_i f(x_i) = \frac{x_n - x_0}{n} \sum_{i=0}^{n} c_i y_i$$

where the c_i are the appropriate coefficients, e.g., for the trapezoidal rule $c_0 = c_n = 1/2$ and $c_i = 1$ otherwise. Let

$$J^* = 1/n \sum_{i=0}^{n} c_i y_i.$$

Then

$$I^* = (x_n - x_0) J^*.$$

Notice that J* does not involve x, and therefore J* can be computed without regard to the scaling of x. For this reason J* rather than I* is obtained by the subroutine.

If n is even Simpson's rule is used throughout the interval (x_0, x_n). If n is odd Simpson's rule is used over the interval (x_0, x_{n-3}) and Newton's three-eighths rule is used over the interval (x_{n-3}, x_n),

therefore

$$J^* = 1/3n \; (y_0 + 4y_1 + 2y_2 + 4y_3 + 2y_4 + \ldots + 4y_{n-1} + y_n) \text{ for n even, and}$$

$$J^* = 1/3n \; (y_0 + 4y_1 + 2y_2 + \ldots + 4y_{n-4} + y_{n-3}) +$$

$$+ 3/8n \; (y_{n-3} + 3y_{n-2} + 3y_{n-1} + y_n) \text{ for n odd.}$$

Error Analysis

Let α be some value in the closed interval (x_0, x_n) and let β and γ be values in the closed intervals (x_0, x_{n-3}) and (x_{n-3}, x_n) respectively.

If $\frac{d^4y}{dx^4}$ is continuous throughout the interval (x_0, x_n) and if n is even

$$\varepsilon = \frac{y^4(\alpha)\ (x_n - x_0)^5}{180n^4}, \text{ where } y^4(\alpha) = \left.\frac{d^4y}{dx^4}\right]_{x=\alpha}$$

If $\frac{d^4y}{dx^4}$ is continuous throughout the invertal (x_0, x_{n-3}) and exists throughout the interval (x_{n-3}, x_n) and if n is odd

$$\varepsilon = 1/n^5 \left[\frac{n-3}{180}\ y^4(\beta) + \frac{3}{80}\ y^4(\gamma)\right]\left[x_n - x_0\right]^5.$$

For the derivation of these quadrature formulas and their error terms, see Milne's Numerical Calculus, pp. 120 thru 124.

Machine Time

The time required for this subroutine is (2.25 + .62n) ms, $n \neq 3$. When n = 3 the time required is 2.73 ms.

Machine Checking

Two preliminary test cases were run:

1. $n = 99$, $y_i = -(2^{35}-1)$ for all i. The result obtained was -2^{35} (it should have been $-(2^{35}-1)$).
2. $n = 98$, $y_i = (2^{35}-1)$ for all i. The correct result, $(2^{35}-1)$, was obtained.

In addition, the following computations were performed:

1. SIN-O was used to produce a table of sines and cosines for the arguments

   $y = \pi X/2 = (\pi/2)\cdot n\cdot 2^{-4}$, n = 0, 1, 2, . . . , 99.

2. Let $S = \int_0^b \cos y dy = \sin b$ and $C = \int_0^b \sin y dy = 1-\cos b$. DIE-O was used to compute S* and C* for $b = (\pi/2)\cdot n\cdot 2^{-4}$, n = 2, 3, 4, . . . 9 ,
   n = 10,15,20,25
   n = 30,40,50, . . . 90, and
   n = 99

3. For each b (or n) $\varepsilon_s = S^* - \sin b$ and $\varepsilon_c = C^* - 1 + \cos b$ were computed. The following tables resulted:

n	$\varepsilon_s \cdot 10^6$	$\varepsilon_c \cdot 10^6$
2	.101	.010
3	.338	.050
4	.197	.039
5	.422	.125
6	.287	.087
7	.490	.215
8	.365	.151
9	.540	.316
10	.429	.229
15	.560	.650
20	.477	.713
25	.200	1.056
30	.101	1.022
40	-.365	.881
50	-.506	.415
60	-.198	.039
70	.286	.087
80	.516	.516
90	.286	.945
99	-.338	.981

It was to be expected that, in general, the errors would be greater for n odd. The overall behavior of ε_s and ε_c is easily seen to be consistent with the fact that ε_s and ε_c represent the errors in integrating the cosine and the sine respectively. For example, ε_s is small for n = 30 because it is the error obtained in

$$\int_0^{\frac{15}{16}\pi} \cos y\,dy = \int_0^{\frac{1}{16}\pi} + \int_{\frac{1}{16}\pi}^{\frac{\pi}{2}} + \int_{\frac{\pi}{2}}^{\frac{15}{16}\pi}$$

$$= \int_0^{\frac{1}{16}\pi} + \int_{\frac{1}{16}\pi}^{\frac{\pi}{2}} - \int_{\frac{1}{16}\pi}^{\frac{\pi}{2}} = \int_0^{\frac{1}{16}\pi}$$

since $\int_{\frac{\pi}{2}}^{\frac{15\pi}{16}} = -\int_{\frac{1}{16}\pi}^{\frac{\pi}{2}}$

In fact, ε_s for n = 30 (b = (15/16)π) is exactly equal to ε_s for n = 2 (b = (1/16)π).

Evidently, all the errors ε_s and ε_c are less than their corresponding maximum estimates as computed by the formulas above under Error estimate. In particular, for n = 10 and 15, the following maximum error estimates were hand computed:

n	Max ε_s 10^6	Max ε_c 10^6
10	.508	.421
15	.740	.903

PX 71900-10-(89)

D		00C00	00013	D	00015	00	00000	00000
D		00T00	00023	I O	00027	00	00000	00000
D		01C00	01078	R R	02066	00	00000	00000
D		00P00	01024	E Y	02000	00	00000	00000
D		00S00	01032	C	02010	00	00000	00000
D		00N00	01059	T	02043	00	00000	00000
D		D1C00	49864	DRUM	63210	00	00000	00000
D		D0P00	49810	STORAGE	63122	00	00000	00000
D		D0S00	49818	DIREC/	63132	00	00000	00000
D		D0N00	49845	TORY	63165	00	00000	00000
D0P00	MS	00000	00P00	NO ALRM EXIT	63122	56	00000	02000
D0P01	MJ	00000	00000	NORMAL EXIT	63123	45	00000	00000
D0P02	TP	A0000	00T01	ENTRY STORE	63124	11	20000	00030
D0P03	TP	Q0000	00T02	N AND ADRS	63125	11	10000	00031
D0P04	DV	01C00	00T03	OF YO	63126	73	02066	00032
D0P05	ZJ	00N00	00P06	IS N EVEN	63127	47	02043	02006
D0P06	TV	00S22	00S19	YES EXIT TO	63130	16	02036	02033
D0P07	TP	00T01	00T03	S20 N/BAR	63131	11	00030	00032
D0S00	MP	01C01	00T01	EQUALS N	63132	71	02067	00030
D0S01	TP	A0000	00T04	STORE 3N	63133	11	20000	00033
D0S02	TP	00C03	00T05	STORE 1	63134	11	00020	00034
D0S03	TP	01C02	00T06	4	63135	11	02070	00035
D0S04	TP	00C03	00T07	AND 1	63136	11	00020	00036
D0S05	TP	00T03	A0000	INDEX IS ONE	63137	11	00032	20000
D0S06	DV	01C00	00T08	HALF N/BAR	63140	73	02066	00037
D0S07	RS	00T08	00C03	MINUS ONE	63141	23	00037	00020
D0S08	TV	00T02	00S11	PRESTORE V	63142	16	00031	02023
D0S09	SP	00C00	00000	CLEAR A	63143	31	00015	00000
D0S10	RP	30003	00S12	FORM	63144	75	30003	02024
D0S11	MA	00T05	00000	SUM	63145	72	00034	00000
D0S12	TP	A0000	00T09	STORE	63146	11	20000	00040
D0S13	LA	A0000	00036	PARTIAL	63147	54	20000	00044
D0S14	TP	A0000	00T00	SUMS	63150	11	20000	00027
D0S15	IJ	00T08	00S23	IS INDEX NEG	63151	41	00037	02037
D0S16	SP	00T09	00036	YES STORE	63152	31	00040	00044
D0S17	SA	00T00	00036	INTEGRAL TO	63153	32	00027	00044
D0S18	DV	00T04	00T04	X N/BAR	63154	73	00033	00033
D0S19	MJ	00000	00000	STORE INTE/	63155	45	00000	00000
D0S20	TP	00T04	A0000	GRAL IN A	63156	11	00033	20000
D0S21	MJ	00000	00P01	GO TO EXIT	63157	45	00000	02001
D0S22	00	00000	00S20	DUMMY/SEE P6	63160	00	00000	02034
D0S23	RA	00S11	01C00	MDFY V BY 2	63161	21	02023	02066
D0S24	SP	00T09	00036	RESTORE PAR/	63162	31	00040	00044
D0S25	SA	00T00	00036	TIAL SUMS	63163	32	00027	00044
D0S26	MJ	00000	00S10	GO TO S10	63164	45	00000	02022
D0N00	TV	00N18	00S19	N IS ODD EX/	63165	16	02065	02033

```
DON01   TP 00T01 A0000        IT TO N5          63166   11 00030 20000
DON02   ST 01C01 00T03        N/BAR IS N/3      63167   36 02067 00032
DON03   ZJ 00S00 00N04        IS N 3            63170   47 02010 02047
DON04   TP A0000 00T04        YES CLEAR T4      63171   11 20000 00033
DON05   LA 00T01 00003          STORE 8N        63172   54 00030 00003
DON06   TP 01C01 00T06          STORE 3         63173   11 02067 00035
DON07   TP 01C03 00T07                9         63174   11 02071 00036
DON08   TP 01C03 00T08                9         63175   11 02071 00037
DON09   TP 01C01 00T09             AND 3        63176   11 02067 00040
DON10   RA 00T02 00T03        STORE ADRS        63177   21 00031 00032
DON11   TV 00T02 00N14        OF YN/BAR         63200   16 00031 02061
DON12   SP 00C00 00000          CLEAR A         63201   31 00015 00000
DON13   RP 30004 00N15        FORM              63202   75 30004 02062
DON14   MA 00T06 00000                  SUM     63203   72 00035 00000
DON15   DV 00T01 00T06        STORE INTE/       63204   73 00030 00035
DON16   RA 00T04 00T06        GRAL IN A         63205   21 00033 00035
DON17   MJ 00000 00P01        GO TO EXIT        63206   45 00000 02001
DON18   00 00000 00N05        DUMMY/SEE NO      63207   00 00000 02050
D1C00   00 00000 00002        CONSTANTS 2       63210   00 00000 00002
D1C01   00 00000 00003                  3       63211   00 00000 00003
D1C02   00 00000 00004                  4       63212   00 00000 00004
D1C03   00 00000 00009              AND 9       63213   00 00000 00011
START
```

THE RAMO-WOOLDRIDGE CORPORATION
Los Angeles 45, California

Interpretive Floating Point Package

Specifications

Identification Tag:	SNAP
Type:	Service Routine (with entrance from program available)
Storage:	Cells (00000-00012) (00724-01023) (70000b-71750b)
	The constant pool is used by this routine.
Service Entrance:	Address 40012b
Program Entrance:	See Description
Coded by:	R. Beach D. Gantner M. Perry M. Speer R. Summers
Approved by:	W. Bauer

October 10, 1955

Description

SNAP is an interpretive floating point package for the ERA 1103. It contains a single address floating binary point arithmetic section, floating decimal data input and output on cards, fixed-to-floating and floating-to-fixed conversion, square root, load and store operations. Additional features include an address modifier, or "B-box", and optional replacement of the result of an operation in the address of the operand. Experience has shown that representative programs using SNAP operate at about 1,000 programmed operations per second. The speed is obtained by fast floating point operations and the fact that many operations are performed at machine speed.

The package is entered by execution of an Interpret (IP) instruction. The u and v portions of this instruction each contain a complete SNAP command. Of these fifteen bits, four contain the operation part, one the B-box option, one the replace option, and the remaining nine the address part. Accordingly, only the lower half (512 cells) of electrostatic may be addressed directly. However, use of the B-box makes any electrostatic or magnetic drum cell indirectly addressable.

A packed number representation is used and results are normalized after each execution. Each floating point number occupies one 36 bit cell. One bit contains the sign, eight the characteristic and twenty-seven the mantissa. Hence the range of the numbers is approximately $\pm 10^{38}$ and approximately eight significant decimal digits are contained in the mantissa. The representation is such that all 1103 logical commands may be validly applied. Therefore, floating point number comparisons may be made at regular machine speed and no need exists for inclusion of comparison commands in the SNAP repertoire.

SNAP occupies approximately 900 cells on the magnetic drum and 300 cells in electrostatic. When a card is to be punched or read, appropriate coding is brought in automatically from the drum. These are the only occasions on which drum references are made.

SNAP commands may be written with alphabetic operations and symbolic addresses, and then assembled by RAWOOP. Floating point numbers may also be included in the program and will be converted and assembled by RAWOOP.

Storage

SNAP uses cells (00000 - 00012), (00724 - 01023) and (70000b - 71750b). In addition, it makes use of the Ramo-Wooldridge constant pool (00013 - 00022). When executing SNAP commands, cell 00000 must contain the word (45 00001 VVVVV).

The following assignments are of interest to the programmer:

00002 F

00003 C

00004 B-box

PX 71900-10-(108)

00005 P_0
00006 P_1
00007 P_2
00008 P_3 } output hopper
00009 P_4
00010 P_5
00011 P_6

Instruction Structure

An instruction occupies a 36 bit cell and contains two SNAP commands, as schematically shown below.

IP	Left Command	Right Command
35-30	29-15	14-0

The left and right SNAP commands are identical in form and have the format shown below.

Operation	B	S	Address
29-26 14-11	25 10	24 9	23-15 8-0

Replace option (result replaced if bit is one)

B-box option (address modified if bit is one)

Number Representation

Floating binary point numbers are each packed into a single 36 bit cell with assignments as follows:

S	Char.	Mantissa
35	34-27	26-0

"Sign" (zero when positive)

The characteristic has a bias of 128 (200b). The binary point lies between bits 26 and 27. To negate a number, the full 36 bits are complemented.

This representation allows use of the Transmit Magnitude and Transmit Negative instructions as well as all comparison jumps. In addition, fixed point decimal output routines give useful conversions when supplied with 27 as a scale factor.

SNAP Commands

The list of commands follows. Parentheses indicate "contents of". F and C are specific electrostatic cells reserved for the package. α is the address part of a SNAP command and may be any address in the lower half of electrostatic (00000-00511). Double length extensions of the results are stored in A, the machine accumulator. The time column below gives average execution times in milliseconds. The operations are further clarified on succeeding pages.

CODE	OPERATION	TIME	RESULT
NO 00	No operation	.43	$(F) \longrightarrow A,F$
AD 04	Add	2.18	$(F) + (\alpha) \longrightarrow A,F$
SU 10	Subtract	2.21	$(F) - (\alpha) \longrightarrow A,F$
MP 14	Multiply	1.94	$(F) \times (\alpha) \longrightarrow A,F$
DV 20	Divide	2.20	$(F) \div (\alpha) \longrightarrow A,F$
PM 24	Poly. Mpy.	3.45	$[(F) \times (C)] + (\alpha) \longrightarrow A,F$
LD 30	Load	1.06	$(\alpha) \longrightarrow A,F$
ST 34	Store	.98	$(F) \longrightarrow \alpha, A,F$
FI 40	Fix	1.39	(F) floating $\longrightarrow$ [A,F] fixed at scale α
FL 44	Float	1.87	(F) fixed at scale α $\longrightarrow$ [A,F] floating
RT 50	Square Root	3.25	$\sqrt{(F)} \longrightarrow A,F$
RD 70	Read Data	500	Floating decimal numbers (four per card), are read, converted, and stored.
PD 74	Punch Data	583	Floating binary numbers are converted and punched as floating decimal numbers (six per card).

NO (00)

The No Operation command is included to complete the right half of a partial instruction. To gain speed, it short-circuits the normal interpretive loop, disabling the replace feature. For this reason the address part (3 octal digits) is available for storage of dummy addresses. When used as a left command, execution time is extended. The B-box option must not be exercised.

AD (04), SU (10), MP (14), DV (20), PM (24).

Results are normalized after each execution. Alarms are provided for "division by zero", and "exponent overflow". When the result is less than 2^{-129}, it is made exactly zero.

LD (30), ST (34)

The Load and Store commands perform functions normally left to standard machine instructions. However, their inclusion in the repertoire extends the use of the B-box, permitting address modification of more complete loops.

Note especially that the reserved cells F, C, B-box, and P_1, are simply electrostatic cells. They may be operated on by machine instructions, as well as SNAP commands. Since the use of machine instructions invariably saves execution time (by a factor of ten to twenty), such use is recommended when storage is not extremely critical or the B-box is not being used.

FI (40), FL (44)

When Fixing or Floating, the scale α is in the standard 1103 notation; i.e. when α is zero, the point is at the extreme right and when α is 35, the point lies between bits a_{35} and a_{34}. The scale α may not be negative but there is no restriction on the upper limit, other than that imposed by overflow. An alarm is provided for this condition.

Since α is not an address, the B-box and store options lose their usual meaning. SNAP does not handle them in the normal manner and they should therefore not be programmed.

RT 50

The square root is accurate to one in the last place. Note that since the argument is in F, the address part may be used (in connection with the replace option) to store the result. An alarm is provided for negative arguments.

RD 70

Upon execution of the Read Data command, floating decimal numbers, each with an associated address (four per card), are read, converted to floating binary numbers, and stored at address α (or α plus B-box) plus the associated address. If it is desired to read in less than four numbers per card, then the associated address field of the decimal number to be ignored must be left blank.

Each floating decimal number consists of a ten digit signed mantissa and a two digit signed exponent. Upon conversion, the mantissa is truncated to 27 bits. The decimal point is at the left of the mantissa field and the leading digit must be non-zero. An alarm is provided if the absolute value of the exponent is greater than 37.

A single RD command initiates the reading of any number of cards. Reading is terminated either by the detection of a blank card, or after loading a card which contains a 12 punch in either column 79 or 80. A blank card or a 12 punch in column 80 returns control to the command following the RD command. A 12 punch in column 79 halts the computer with PAK = 71336 after the card has been loaded. Starting will then return control to the command following the RD command. Reading is accomplished at the rate of 480 numbers per minute.

The following card column assignments are made:

(1-4)(22-23)(41-42)(60-61)	Not used
(5-9)(24-28)(43-47)(62-66)	Associated address
(10-19)(29-38)(48-57)(67-76)	Mantissa (decimal point at left)
(20-21)(39-40)(58-59)(77-78)	Exponent of 10
(79-80)	12 punch will terminate read.

If the exponent or the mantissa is negative, an eleven punch is included over its rightmost digit.

PD (74)

Output is in floating decimal, eight significant digits with sign and a signed exponent whose range is $\pm$ 38. Results may be punched in any of six punch positions, up to six numbers per card. In addition, a four digit identification number may be punched. Punching may occur at rates up to 100 cards per minute.

Dependent on the value of α, execution of the command (PD α) stores the number for future punching or punches a card. Specifically, when $0 \leq \alpha \leq 6$, (F) is stored in cell P_α for future punching in card field α. When α is 10, 20, 30, or 50, contents of the output hopper are converted to decimal and punched, after which the output hopper is set to zero. Only the four least significant octal digits of the U-address part of P_0 are converted to a decimal integer and punched in the identification field. If (P_i) is zero, its field in the output card will be blank. All SNAP output cards have a 12 punch in column 75 and also have a 12 punch in column 78, 79, or 80 when α is 50, 30, or 20 respectively. The SNAP output board is wired to give single spacing before printing except when 78, 79, or 80 contain 12 punches. In these cases, page ejection, triple spacing, or double spacing occur respectively, before printing.

Because of the characteristics of the Bull Reproducer, a blank card is pushed through the punch side on the first read cycle following a punch cycle. SNAP keeps track of these sequences and positions an extra card when required. Since a card cycle of one half second is wasted each time this occurs, interspersed reading and punching should be avoided when convenient. The blank cards may be removed from the output deck by a single sort, selecting 12 punches in column 75, since only legitimate output cards contain this punch. Upon completion of a program using the PD command, the Bull should be cleared manually to punch and feed the final card.

PX 71900-10-(108)

The output hopper (cells 00005-00011) need not be loaded by a PD command, but may be loaded by any appropriate 1103 or SNAP command. It is also possible to use the hopper as temporary storage. Note, however, that if fixed point numbers remain in the hopper when punching is initated, the punch routine may fail.

The following card column assignments are made:

(1)	(13)	(25)	(37)	(49)	(61)	Sign of mantissa
(2-9)	(14-21)	(26-33)	(38-45)	(50-57)	(62-69)	Mantissa (point at left)
(10)	(22)	(34)	(46)	(58)	(70)	Sign of exponent
(11-12)	(23-24)	(35-36)	(47-48)	(59-60)	(71-72)	Exponent
(77-80)						Identification

Replace Option

When a bit is present in the replace position of a SNAP command, the result of the execution is stored (replaced) at the effective address of the operation. This is in addition to the normal storage of the result in A and F.

B-box Option

The B-box, or address modifier, is contained in cell 00004. When a SNAP command has a bit present in its B-box position, the U-portion of cell 00004 is added to the address part (a) of the command to form the effective address for this execution. The operation part and the V-portion of the B-box must be zero (either positive or negative), and the effective address must not exceed 77777b. Use of this option does not alter the actual address which remains constant.

The B-box is loaded by standard 1103 Transmit instructions and may be incremented or decremented by the Replace instructions.

If the effective address exceeds 77777b, a portion of SNAP will be destroyed. Note that, although the nine bit address part of a SNAP command compels the actual address to be less than 512, the effective address using the B-box may be as large as 77777b. Note also that it is possible to decrement an actual address by loading the B-box with a negative number.

Symbolic Coding

Although it is possible to code for SNAP in octal, it is much easier to code symbolically for assembly on RAWOOP. The advantages of using alphabetic operations and symbolic addresses are magnified by the peculiar command structure of SNAP instructions. Regular machine instructions may be interspersed arbitrarily with SNAP instructions.

At the end of this section is a facsimile of a standard RAWOOP coding sheet. It contains some sample floating point numbers and SNAP instructions (not a program), and their assembled equivalents (in the Comments column). There are two alphabetic operations, two symbolic addresses, and two optional B-box and store columns in each instruction. The left hand items in each pair are combined to form the left SNAP command, the right hand items to form the right command. The numbers in rows (1), (2), and (3) are respectively .125 -4096, and 1.125.

	ADDRESS	OP	U	V	±	D	±	B	COMMENTS
	D		99Z00	256					
(1)		1	25		-	01	F		17 64000 00000
(2)		-4	096			3	F		56 23777 77777
(3)		1	125				F		20 14400 00000
(4)		LDSU	99Z02	99Z04		B		BS	14 32402 13404
(5)		ADFI	99Z03	21		BS			14 07403 40025
(6)		RDLD	99Z00	99Z05					14 70400 30405
(7)		ADNO	99Z07			S			14 05407 00000

Consult the RAWOOP write-up for more complete details on symbolic coding.

Alarms

An alarm routine is self-contained in SNAP. When an alarm condition is reached, the flexowriter types out a word having the form

TT P-XXXXX

where TT is the alarm type, P is an L or an R (left or right command), and XXXXX is the location of the instruction which caused the alarm. Upon completion of the typing, a Manual Stop zero occurs. If the computer is now started, F and A are set to zero and execution of the next command is initiated. See below for the exception when reading data.

The following types of alarm may occur:

1. EO Exponent Overflow - The absolute value of the result is greater than $(1 - 2^{-27})2^{126}$.

2. RT Square Root of a negative number

3. DV Division by zero

4. FI Fixing causes the number to overflow the F register

5. RD Absolute value of input number is equal to or greater than 10^{38}. Starting the computer after alarm causes normal reading except that a zero replaces the erroneous number.

Operating Instructions (Activation of SNAP)

1. Insure that SNAP is intact on the magnetic drum. (This can be accomplished by a transfer of the Service Routine Library from magnetic tape).

2. Load the problem program - Program should include the R-W constant pool.

PX 71900-10-(108)

3. Set PAK to 40012b and start - This causes SNAP to be read into its electrostatic locations, sets the B-box, F, C, and output hopper to zero, supplies an appropriate jump in cell zero, positions cards on each side of the reproducer, and gives control to cell 40000 which normally initiates execution of the problem program.

Programmed Activation of SNAP

It is occasionally desirable to activate SNAP from the program. This may be accomplished by execution of one of the two commands given below.

It is assumed that cell 40000b contains (MJ 00000 VVVVV) which RAWOOP supplies.

1. For initial activation (37 40000 40012) - execution of this instruction has the same effect as (3) in Operating Instructions above except that control is given to cell (n + 1) when the instruction is in cell n.

2. For reactivation
   (a) (37 40000 71050) - same as (1) above except that card positioning is omitted.
   (b) (37 40000 71047) - same as (2,a) above except that one blank is fed on the punch side to insure that a RD has not cleared the punch station.

Subroutines

Many of the common subroutines have been coded using the SNAP number representation for argument and result. Others are in process. Their use is extremely simple.

It is, however, possible to employ any fixed point subroutine in a floating point program by the expedient of the Fix and Float commands contained in SNAP. Note that conversion is rapid and that scaling is easily supplied in the address part of the command.

Design Criteria

The philosophy which leads to this package states that, in order of importance, the following three items are virtues:

(1) Simplicity - To produce faster coding with fewer errors.

(2) Execution speed - To conserve machine time.

(3) Minimum storage usage - Both in the problem program and in the package, to conserve high speed storage.

With these in mind, a single address system was chosen. While logical operations are generally best served by multiple addresses, use of a single address is indicated for arithmetic operations. Since the four basic arithmetic operations may occur in a great variety of sequences, a multiple address system must either waste many of the addresses or contain contain a very large number of arithmetic command types. Wasting addresses wastes storage in the main program; inclusion of a large number of command types expends storage in the package. Thus (3) is satisfied by the single address. With only one kind of each of the arithmetic operations, jumping and sorting in the arithmetic section is minimized.

Furthermore, use of only one operand saves the unpacking of multiple packed numbers. So the requirements of (2) are met. Finally, the consideration of (1), ultimate simplicity is achieved with a system which always operates on the same register with but a single operand.

SNAP Sampler Trace

Description

This routine monitors the course of a SNAP program by punching out the results of those SNAP commands which are specified in a list prepared by the programmer. A parameter word will indicate the location of this list.

This list has the following specifications:

a. The list is made up of sublists of four words each. These sublists have the form:

00	FA	LA
00	Np	Ns
00	00000	00000
00	00000	00000

FA is the address where the trace is to start and LA the address at which it is to stop. Np is the number of times FA is to be passed before starting the trace, while Ns is the number of times the section FA to LA is to be traced. The last two words are used by the trace to store the blocked instructions.

b. This list must not be placed in cells 15b thru 107b, but may be put on the drum. In any case, a parameter word 00 Lo Lf will specify its location. Lo is the address of the first word of the list and Lf is the address of the last word. This parameter word must be loaded into cell 71777b Restoring the library from magnetic tape loads an all zero word into cell 71777b. If this word is not changed, a complete trace of all SNAP commands is automatically performed

c. Any number of sublists may be used. A particular address must appear only once in the list since blocking a blocked instruction is not possible.

d. The storage addresses FA and LA must be the addresses in which the instructions to be blocked are actually stored at the time the blocking routine is activated.

The operation of the trace is as follows:

a. When a blocked FA is reached in the program, N_p and N_s are examined. If $N_p = 0$, $N_s \neq 0$ then tracing is initiated and a start indicator placed in the word containing N_p and N_s. On the other hand, if $N_p = 0$ and $N_s \neq 0$ does not occur, then no action is taken.

b. At each SNAP execution, after a trace start, the F register is transferred to the next available position in the trace output hopper (last seven cells in ES). When this hopper contains the results of six SNAP commands (three IP instructions) then the SNAP output routine is used to punch a card containing the information in the trace hopper. The identification field of the output card contains the address of the instruction which produced the numbers in fields one and two. This address is in octal and will be 1725 if the instruction was an FA. The SNAP output command operation is in no way altered during tracing.

c. To empty the trace hopper at any time, start at 72125b. One card will be punched and the machine will stop on MS 0 with PAK = 77777b. SNAP must be in ES to exercise this option.

d. In the event the trace hopper is emptied when it contains no information, a card will be punched containing the address 7777 in the identification field, while the rest of the card will be blank.

e. When a blocked LA is reached in the program and if a start indicator was set up in the word containing the associated N_p and N_s then tracing is stopped and the trace hopper emptied. Otherwise, no action is taken.

f. The execution of the instruction at FA will be traced, but that at LA will not.

PX 71900-10-(140)

g. The seven cells of ES, 1771b - 1777b, cannot be used by the programmer if the trace is to be employed. Normal SNAP operation will not destroy the contents of these cells.

Programming Instructions

1. Load cell 71777b with the parameter word 00 Lo Lf where Lo and Lf are the locations of the first and last words of the list.
2. Load the list in form described above.
3. Start at 72000b. The routine will block the proper instructions as per the list supplied, modify SNAP in order to perform the trace, and then stop with PAK set at 40012b.
4. In the event no parameter word is loaded (and no list is supplied) a start at 72000b will initiate the blocking routine to modify SNAP on MD so that all SNAP commands will be traced. A stop will follow with PAK set at 40012b.

Warnings and Restrictions

1. The list can not occupy cells 15b - 107b.
2. Cells 1771b - 1777b must be reserved for the trace hopper and cannot be used by the programmer.
3. SNAP must be in ES at the time each FA and LA is reached in the program.
4. The instructions in FA and LA must not be read into or out of by the program.
5. Only SNAP commands are traced.
6. The trace can only be used with SNAP and not with the complex version.
7. Activating the trace modifies the copy of SNAP on MD and destroys the complex arithmetic portion of SNAP. To start another program using SNAP or its complex version, it is necessary to restore the library from magnetic tape.

8. An abnormality exists for the following type of a list:

00	FA_1	LA_1
00	N_{p1}	N_{s1}
00	00000	00000
00	00000	00000
00	FA_2	LA_2
00	N_{p2}	N_{s2}
00	00000	00000
00	00000	00000

Assume that N_{p1}, N_{s1}, N_{p2}, N_{s2} are such that tracing in both sublists is concurrent. Further, suppose that FA_1, FA_2, LA_2, LA_1 are executed in the order given. Hence, the trace will be initiated at FA_1 and once again at FA_2. When LA_2 is reached, the trace will stop since it was started at FA_2.

The instructions from LA_2 to LA_1 will not be traced and the trace will be stopped once again when LA_1 is reached. At this time the hopper will be punched. Other unusual combinations can be analyzed in a similar fashion.

PX 71900-10-(140)

THE RAMO-WOOLDRIDGE CORPORATION
Los Angeles 45, California

Interpretive Floating Point Package - Complex

Identification:	SNIP
Type:	Service Routine (with entrance from program available)
Storage:	Cells
	634 thru 1023
	70000b thru 71662b This includes SNAP
	The constant pool is used by this routine.
Service Entrance:	Address 40013b
Program Entrance:	See description

Coded by:	C. Koos	
	M. Perry	January, 1956
Code Checked by:	C. Koos	January, 1956
Machine Checked by:	C. Koos	January, 1956
Approved by:	W. F. Bauer	April, 1956

Description

SNIP is the complex arithmetic version of SNAP, a floating point interpretive package. An understanding of the use of SNAP is presupposed.

The activation of this routine changes SNAP into SNIP on the magnetic drum. The original version of SNAP can be obtained again only by a transfer of the Service Routine Library from magnetic tape.

SNIP performs its operations in either real or complex arithmetic depending on a mode which is selected by the programmer, and may be changed at any time.

In the complex mode

1. The complex numbers to be operated on must be in rectangular form, with the real and imaginary parts of the number stored in consecutive cells (For example, the complex number x + iy would be stored in the machine with x in cell α and y in cell $\alpha + 1$).

2. The floating Complex Accumulator requires two cells: Cell 00002, F, is used for the real part; Cell 00003, C, is used for the imaginary part, that is, the two cells 00002 and 00003 constitute the Complex Floating Accumulator, F_c.

3. The Polynomial Multiply command of SNAP is changed so that its execution will result in computing the absolute value of the number stored in F_c.

4. The Fix, Float, Read, Punch and No Operation commands operate exactly as in SNAP, while the remaining commands are changed only in the sense that they now use both cells 00002 and 00003 for the floating accumulator and cells α and $\alpha + 1$ for the argument as explained above.

5. The machine accumulator A contains the real part of the result after the execution of any one of these operations.

6. The Replace and B-box options may be used in all cases that are permitted by SNAP, with two consecutive cells being operated on as described. Keep in mind that the B-box must be indexed by two when used in referencing a list of complex numbers.

7. It is not permissible to load F_c with TP instructions; a Load command must be executed for this purpose.

In the real mode

1. All SNAP commands except Polynomial Multiply operate as in SNAP itself.

2. The execution of the Polynomial Multiply command will give the absolute value of (F_c) just as it does in the complex mode.

PX 71900-10-(141)

SNIP commands

The Complex Accumulator, F_c, is defined as two specific electrostatic cells which contain the complex number x + iy: cell 00002, F, contains x and cell 00003, C, contains y. Both x and y are stored as SNAP numbers; that is, each has its own binary exponent. The notation α_c represents the address of a complex number x + iy, where x is stored at α and y at $\alpha + 1$, that is, $(\alpha) = x$, $(\alpha + 1) = y$.

The following definitions apply when in the complex mode:

Code	Result
AD 04	$(F_c) + (\alpha_c) \rightarrow F_c$; $(F) \rightarrow A$
SU 10	$(F_c) - (\alpha_c) \rightarrow F_c$; $(F) \rightarrow A$
MP 14	$(F_c) \times (\alpha_c) \rightarrow F_c$; $(F) \rightarrow A$
DV 20	$(F_c) \quad (\alpha_c) \rightarrow F_c$; $(F) \rightarrow A$
PM 24	$\lvert(F_c)\rvert \longrightarrow F_c$; $(F) \rightarrow A$
LD 30	$(\alpha_c) \longrightarrow F_c$; $(F) \rightarrow A$
ST 34	$(F_c) \longrightarrow \alpha_c$; $(F) \rightarrow A$
RT 50	$\sqrt{(F_c)} \longrightarrow F_c$; $(F) \rightarrow A$

The NO (no operation), FI (fix), FL (float), RD (read data) and PD (punch data) instructions of SNAP are unaltered. The PM (polynomial multiply) instruction of SNAP is replaced by the absolute value instruction whether operating in the real or complex arithmetic mode if SNIP has been activated.

Manual Activation of SNIP

1. Insure that both this routine and SNAP are intact on the magnetic drum. (This can be accomplished by a transfer of the Service Routine Library from magnetic tape).

2. Load the problem program - The program should include the Ramo-Wooldridge constant pool, and a jump to start in 40000b, as supplied by RAWOOP.

3. Set PAK to 40013 and start--this changes SNAP into SNIP and causes it to be read into its electrostatic locations, sets the B-box, F, C, and output hopper to zero, supplies an appropriate jump in cells zero and one, positions cards on both sides of the reproducer, and gives control to cell 40000b which normally initiates execution of the problem program. At this time the routine is in the <u>real</u> arithmetic mode.

Programmed Activation of SNIP

Depending on the card positioning desired any one of three different return jump instructions may be used to activate SNIP from the program. Each assumes that there is a manual jump instruction in cell 40000b (such as that supplied by RAWOOP), and in each case control is returned to the instruction immediately following the return jump.

a. Execution of 37 40000 40013b positions cards on both sides of the reproducer.

b. Execution of 37 40000 71644b feeds one card on the punch side of the reproducer.

c. Execution of 37 40000 71646b omits all card positioning.

Otherwise the effect of programmed activation is the same as that described in step 3 under manual activation.

Switching Modes

Activation of SNIP by any one of the methods described above leaves it in the real arithmetic mode. At any time after SNIP has been activated the mode may be switched as follows:

To switch from the real mode to the complex mode execute the return jump

37 01541 01713b

To switch from the complex mode to the real mode execute the return jump

37 01541 01715b

In either case the desired mode change is accomplished, cell 00003, C, is set to zero, and control is returned to the cell immediately following the return jump. The real mode should ordinarily be used wherever possible because it is considerably faster than the complex mode.

Alarms

The SNAP alarm routine is used, with the possibility of the same type of alarm occurring (EW, RT, DV, FI, RD). It is not advisable to continue the problem after an alarm, since either the real or imaginary part of a number may have caused the alarm.

"A DV alarm will occur if the real and the imaginary parts of the denominator are both less than about $2 \cdot 2^{-65}$ in absolute value.

PX 71900-10-(141)

THE RAMO-WOOLDRIDGE CORPORATION
Los Angeles 45, California

The Ferranti Input Routine (revised)

Specifications

Identification Tag:	FRI-0
Type:	Service routine (with subroutine entrance)
Special Storage:	The constant and temporary storage pools are not used by this routine
Service Entrance:	Address 40001b
Program Entrance:	Address 40001b
Program Exit:	Address 40020b
Alarm Exit:	The alarm exit is used by this routine

Coded by:	R. Beach	May 18, 1955
Code Checked by:	R. Summers	May 19, 1955
Machine Checked by:	R. Beach	August 4, 1955
Revised by:	C. Koos	December 1, 1955
Approved by:	W. Bauer	December 9, 1955

Description

I. General

This routine is designed to read, by means of the Ferranti reader, seven-level bioctal tape prepared as described below. The routine reads in paper tape at the full speed of the Ferranti with only short hesitation when a check or insert address is encountered.

If desired, the tape may contain a check sum to be tested for agreement with the computed sum of the data read-in. The routine will read data into any ES or MD cell although the reading of information into certain drum cells (as described in detail below) will result in abnormal operation.

The routine stores the contents of ES on MD at addresses 76000b through 77777b and then transfers itself to ES. It sums itself (in ES) and checks the sum against the correct sum (stored on MD).

The Ferranti reader is started in the free running mode and the routine proceeds to read tape and process the information contained on the tape in the same manner as does the ERA photoelectric reader (for exceptions, see II. 3 and 4).

Each word to be transferred to memory is summed as it is read in from tape. Words which are to be read into ES are first stored in the MD image of ES (76000b thru 77777b).

During operation all words are read into ES from the tape and a block transfer to MD is made when (1) ES has been filled with data (that is, when 924 words have been read in); (2) an insert address appears on the tape; or (3) the "end of tape" seven-level combination has been read in (see II. 4).

The reader is stopped before making the transfer and is started again after the transfer has been completed in the first two cases; in the last case, the reader is stopped, ES is restored from the MD image and control is transferred to the exit.

The reader is also halted when a check address appears on the tape. If no check sum test (see II. 3) is to be made after a successful check address test the reader is started immediately; if the check sum test is specified the reader is started after the test is made and the sum determined to be correct.

The routine does not prevent read in to addresses 76000b thru 77777b nor to those cells used by the routine for its own operations.

II. Requirements for Tape Preparation

1. The first word on a tape must be an insert address.

2. Check addresses should be used, although FRI-0 will operate without them. A check address immediately following an insert address must be the same as the insert address.

PX 71900-10-(63)

3. For a check sum test the following four words must appear on the tape at the point where the sum is to be tested:

   a. Insert address 75202b

   b. High order 36 bits of check sum

   c. Low order 36 bits of check sum

   d. Check address 75204b

Operating Instructions (to be followed when the routine is used as a service routine)

1. Set PAK to 40001b and start.

2. Computation will halt with the MS instruction 56 00000 40001b at the completion of the read in.

Programming Instructions (to be followed when the routine is used as a subroutine)

1. Enter the routine with the RJ instruction 37 40020 40001b

2. Control is returned to the cell immediately following the RJ instruction as soon as an "end of block" punch is reached on the tape.

Alarm Conditions

1. No "end of tape" punch. This condition is indicated bby the tape running completely out of the Ferranti reader. When such a condition occurs the operator should.

   a. Master clear

   b. Set PAK to 00074b and start

   c. When computation halts (when a service entry was used) with the MS instruction 56 00000 40001b the machine will be returned to its original state and the data read from the tape will be properly stored.

   If a program entry was used control will be transferred to the proper cell in the main program.

2. FRI-0 not transferred to ES correctly. If ALR-1 prints "FRI-0 xxxxx and (A) and (Q), the sum of the program transferred to ES has failed to check. Starting at this point transfers FRI-0 to ES again.

   A second failure indicates that FRI-0 is not on the drum correctly and should be restored.

3. Check address failure. If ALR-1 prints "ALAR C" and (A) and (Q), a check address has failed. In the alarm print (A_R) is the address of the next

cell to be loaded and (Q) is the check address that was read in from paper tape.

Starting at this time will cause the machine to ignore the failure and operation will continue normally.

4. Check sum failure. If ALR-1 prints "ALAR M" and (A) and (Q), the check sum on the tape has failed to agree with the computed sum. The computed sum is in A at the time the alarm print occurs.

   Starting at this point will cause the routine to ignore the failure and to begin to read in the tape again.

   If at any time (ES) need to be restored from its image, starting at 40040b will transfer the image to ES and transfer control to the FRI-0 exit.

5. And "end of tape" (or "end of block") punch must be present on the tape to halt read in. This consists of seventh level punches in two consecutive frames on the tape at the point where the read in is to be stopped. This seventh level combination acts as a signal to FRI-0 to restore (ES) and stop the Ferranti reader. It is compatible with the ERA photoelectric reader in that it is an illegal combination which halts the ERA reader.

   Note: If there is a sixth level punch in the second of two consecutive frames having seventh level punches the stop is bypassed. The check sum is cleared and the reading continues. This will still be an illegal combination which will halt an ERA photo-electric reader.

PX 71900-10-(63)

THE RAMO-WOOLDRIDGE CORPORATION
Los Angeles 45, California

Arcsine-Arcosine Routine

Specifications

Identification Tag:	SNI-1
Type:	Subroutine
Assembly Routine Spec:	SUB 50410 08014
Storage:	80 words total program storage
	7 words temporary storage pool used, addresses 00027b thru 00035b
	The constant pool is used by this routine
Entrance and Exit:	RJ 00K01 00K03 for the arcsine
	RJ 00K01 00K02 for the arcosine
Alarm:	The alarm exit is used to print "Alarm" if the absolute value of the argument exceeds one.
Drum Assignment:	Addresses 64252b thru 64371b
Machine Time:	6.0 ms average, 6.6 ms maximum time
Mode of Operation:	Fixed point

Coded by:	A. Franck (ERA)	May 14, 1955
Translated by:	D. Gantner	August 16, 1955
Machine Checked by:	T. Tack	August 25, 1955
Approved by:	W. Bauer	September 12, 1955

Description

This subroutine computes F(x) = arcsin x or F(x) = arcos x (depending on which of two entrances is used) by use of a polynomial approximation. (See Rand Sheet No. 39).

The routine was originally coded by Dr. A. Franck of ERA and has been adopted for use at The Ramo-Wooldridge Corporation.

Notation

x = sine or cosine of an angle F(x).

F(x) = the computed angle in radians whose sine or cosine is x.

The ranges of the results are the principal values, defined as follows:

$$-\pi/2 \leq \text{arcsine } x \leq \pi/2$$

$$0 \leq \text{arcosine } x \leq \pi$$

Programming Instructions

1. Place the argument scaled by 2^{33} (i.e. $x \cdot 2^{33}$) in A.

2. Enter the routine with an RJ instruction. If the routine was assigned to region 00K00 for assembly use the instruction

   RJ 00K01 00K03 for the arcsine, or

   RJ 00K01 00K02 for the arcosine.

3. At the time of exit from the routine the result $F(x) \cdot 2^{33}$ is left in A.

Alarm Conditions

If the $|x|$ is greater than one an alarm exit will occur. The word "alarm" and the address of the cell in the main program containing the RJ instruction which was used to enter SNI-1 will be printed on the flexowriter.

Pushing the start button after the alarm halt will transfer control to the exit of SNI-1.

PX 71900-10-(148)

D		DSC00	50410			64252	00	00000	00000
D		ASC00	01024			02000	00	00000	00000
DSC00	37	75701	75702	B	ALARM EXIT	64252	37	75701	75702
DSC01	MJ	00000	00000		ROUTINE EXIT	64253	45	00000	00000
DSC02	MJ	00000	ASC57		ARCCOS ENTRY	64254	45	00000	02071
DSC03	MJ	00000	ASC60		ARCSIN ENTRY	64255	45	00000	02074
DSC04	TP	00013	00024			64256	11	00015	00030
DSC05	TM	A0000	00025			64257	12	20000	00031
DSC06	TJ	ASC66	ASC00			64260	42	02102	02000
DSC07	TJ	ASC67	ASC09			64261	42	02103	02011
DSC08	MJ	00000	ASC00			64262	45	00000	02000
DSC09	TP	A0000	Q0000			64263	11	20000	10000
DSC10	ZJ	ASC11	ASC52			64264	47	02013	02064
DSC11	SJ	ASC12	ASC13			64265	46	02014	02015
DSC12	TP	00016	00024			64266	11	00020	00030
DSC13	TM	A0000	A0000			64267	12	20000	20000
DSC14	EJ	ASC68	ASC55			64270	43	02104	02067
DSC15	MP	00025	ASC69			64271	71	00031	02105
DSC16	LA	A0000	00037			64272	54	20000	00045
DSC17	AT	ASC70	00026			64273	35	02106	00032
DSC18	MP	00025	00026			64274	71	00031	00032
DSC19	LA	A0000	00037			64275	54	20000	00045
DSC20	AT	ASC71	00026			64276	35	02107	00032
DSC21	MP	00025	00026			64277	71	00031	00032
DSC22	LA	A0000	00039			64300	54	20000	00047
DSC23	AT	ASC72	00026			64301	35	02110	00032
DSC24	MP	00025	00026			64302	71	00031	00032
DSC25	LA	A0000	00038			64303	54	20000	00046
DSC26	AT	ASC73	00026			64304	35	02111	00032
DSC27	MP	00025	00026			64305	71	00031	00032
DSC28	LA	A0000	00038			64306	54	20000	00046
DSC29	AT	ASC74	00026			64307	35	02112	00032
DSC30	MP	00025	00026			64310	71	00031	00032
DSC31	LA	A0000	00038			64311	54	20000	00046
DSC32	AT	ASC75	00026			64312	35	02113	00032
DSC33	MP	00025	00026			64313	71	00031	00032
DSC34	LA	A0000	00036			64314	54	20000	00044
DSC35	AT	ASC76	00026			64315	35	02114	00032
DSC36	TN	00025	A0000			64316	13	00031	20000
DSC37	SA	ASC68	00002			64317	32	02104	00002
DSC38	TP	ASC78	00027			64320	11	02116	00033
DSC39	EJ	ASC78	ASC47			64321	43	02116	02057
DSC40	TP	A0000	00028			64322	11	20000	00034
DSC41	SP	00028	00034			64323	31	00034	00042
DSC42	DV	00027	00029			64324	73	00033	00035
DSC43	LA	00027	00071			64325	54	00033	00107
DSC44	RS	Q0000	00027			64326	23	10000	00033

DSC45	RA	00027	00029		64327	21	00033	00035
DSC46	QJ	ASC41	ASC47		64330	44	02051	02057
DSC47	MP	A0000	00026		64331	71	20000	00032
DSC48	LA	A0000	00037		64332	54	20000	00045
DSC49	ST	ASC77	Q0000		64333	36	02115	10000
DSC50	IJ	00024	ASC52		64334	41	00030	02064
DSC51	TN	Q0000	Q0000		64335	13	10000	10000
DSC52	RS	Q0000	00023		64336	23	10000	00027
DSC53	DV	ASC79	A0000		64337	73	02117	20000
DSC54	MJ	00000	ASC01		64340	45	00000	02001
DSC55	TN	ASC77	Q0000		64341	13	02115	10000
DSC56	MJ	00000	ASC50		64342	45	00000	02062
DSC57	TP	ASC77	00023		64343	11	02115	00027
DSC58	TV	ASC65	ASC54		64344	16	02101	02066
DSC59	MJ	00000	ASC04		64345	45	00000	02004
DSC60	TP	00013	00023		64346	11	00015	00027
DSC61	TV	ASC64	ASC54		64347	16	02100	02066
DSC62	MJ	00000	ASC04		64350	45	00000	02004
DSC63	TN	A0000	A0000		64351	13	20000	20000
DSC64	MJ	00000	ASC01		64352	45	00000	02001
DSC65	00	00000	ASC63		64353	00	00000	02077
DSC66	67	77777	77777	B	64354	67	77777	77777
DSC67	10	00000	00001	B	64355	10	00000	00001
DSC68	10	00000	00000	B	64356	10	00000	00000
DSC69	53	24135	20070	B	64357	53	24135	20070
DSC70	33	24414	25535	B	64360	33	24414	25535
DSC71	56	40071	51545	B	64361	56	40071	51545
DSC72	37	50417	41233	B	64362	37	50417	41233
DSC73	46	23706	66522	B	64363	46	23706	66522
DSC74	26	61651	66073	B	64364	26	61651	66073
DSC75	44	42003	30653	B	64365	44	42003	30653
DSC76	31	10375	51633	B	64366	31	10375	51633
DSC77	31	10375	52421	B	64367	31	10375	52421
DSC78	37	77777	77777	B	64370	37	77777	77777
DSC79	00	00000	00002	B	64371	00	00000	00002
START								

PX 71900-10-(148)

THE RAMO-WOOLDRIDGE CORPORATION
Los Angeles 45, California

Floating Point Arcsine-Arccosine Routine

Specifications

Identification Tag:	SNI-2
Type:	Subroutine
Assembly Routine Spec:	SUB 50349 06112
Storage:	61 words total program storage. 3 words temporary storage pool used, addresses 00027b thru 00031b The constant pool is used by this routine.
Entrance and Exit:	RJ 00K01 00K02 for the arcsine RJ 00K01 00K03 for the arcosine
Alarm Exit:	The alarm exit is used to print "alarm" if the absolute value of the argument exceeds one.
Drum Assignment:	Address 64155b thru 64251b
Machine Time:	7.17 ms average, 8.74 ms maximum
Mode of Operation:	Floating point

Coded by:	M. Perry	August 25, 1955
Code Checked by:	R. Bigelow	August 28, 1955
Machine Checked by:	M. Perry	September 7, 1955
Approved by:	W. Bauer	September 12, 1955

Description

When supplied with an argument X in SNAP form, this routine will compute the arcsine or the arcosine of X (depending on which of two entrances was used) using a Rand Polynomial Approximation producing the answer in SNAP form.

Programming Instructions

The SNAP floating point routine must be in E.S. when this subroutine is entered.

1. Place the argument X in the accumulator. X must be in SNAP form.

2. Return Jump to the subroutine. Assuming that the subroutine was assigned to region 00K00 for assembly, use the instruction RJ 00K01 00K02 for arcsine, or RJ 00K01 00K03 for arcosine.

3. At the time of exit from the subroutine, the double length extension of arcsine X, or arcosine X, in SNAP form will be in the accumulator.

The ranges of the results are the principal values, defined as follows:

$$-\pi/2 \leq \text{arcsine } X \leq \pi/2$$
$$0 \leq \text{arcosine } X \leq \pi/$$

Error Analysis

The error in the result produced by this subroutine is less than 2^{-26}.

Mathematical Analysis

1. The Rand Polynomial Number 39 is evaluated using the absolute value of X as the argument. Designate the result as P(X).

2. The square root of 1 minus the absolute value of X is found using the square root subroutine within SNAP. Designate this result as R(X).

3. If X is positive, let $Y = P(X)R(X)$

   If X is negative, let $Y = \pi - P(X)R(X)$

4. Arcsine X $= (\pi/2) - Y$

   Arcosine X $= Y$

5. This procedure places arcsine X in the first or fourth quadrant, and arcosine X in the first or second quadrant.

Alarm Conditions

An alarm print will occur if the argument is outside the range $-1 \leq X \leq 1$. The flexowriter will print "alarm" and the address of the cell in the main program containing the RJ instruction which was used to enter SNI-2.

PX 71900-10-(149)

D		00K00	00906		SNAP CONS		01612	00	00000	00000
D		00T00	00722		SNAP TEMP		01322	00	00000	00000
D		00C00	00754		SNAP EXIT		01362	00	00000	00000
D		SRT00	00927		SNAP SQ ROOT		01637	00	00000	00000
D		00P00	00877		SNAP CONS		01555	00	00000	00000
D		0AS00	50349		ARCSN RTN	49	64155	00	00000	00000
D		1AS00	01024		REL 2000	49	02000	00	00000	00000
D		2AS00	50398		ARCSN CNS	11	64236	00	00000	00000
D		3AS00	01073		REL 2000	11	02061	00	00000	00000
0AS00	37	75701	75702	B	ALARM EXIT		64155	37	75701	75702
0AS01	MJ				NORMAL EXIT		64156	45	00000	00000
0AS02	RP	20002	1AS04				64157	75	20002	02004
0AS03	TP	00018	00023		COS ENTRY		64160	11	00022	00027
0AS04	TP	A0000	00024		X		64161	11	20000	00030
0AS05	TM	A0000	A0000		X ABS		64162	12	20000	20000
0AS06	TP	B0000	00025		CLEAR 25		64163	11	30000	00031
0AS07	ST	3AS08	Q0000		X-50		64164	36	02071	10000
0AS08	TJ	00K14	1AS12		X ZERO		64165	42	01630	02014
0AS09	QT	00K03	00025		M		64166	51	01615	00031
0AS10	SP	Q0000	00008				64167	31	10000	00010
0AS11	TV	B0000	1AS12				64170	16	30000	02014
0AS12	LA	00025	00000		M FIXED	33	64171	54	00031	00000
0AS13	ST	3AS09	Q0000		X LESS 1		64172	36	02072	10000
0AS14	SF	Q0000	00T04				64173	74	10000	01326
0AS15	SJ	1AS17	1AS16				64174	46	02021	02020
0AS16	ZJ	1AS00	1AS32				64175	47	02000	02040
0AS17	LA	A0000	00027				64176	54	20000	00033
0AS18	TN	B0000	00T03		1/X FLOATED		64177	13	30000	01325
0AS19	LA	00T04	00027		SF IN T04		64200	54	01326	00033
0AS20	TP	1AS01	00C00				64201	11	02001	01362
0AS21	RJ	00C00	SRT00		FIND ROOT		64202	37	01362	01637
0AS22	TP	00P02	00C00		REPAIR SNAP		64203	11	01557	01362
0AS23	SP	00T04	00000				64204	31	01326	00000
0AS24	SS	SRT26	C0008				64205	34	01671	00010
0AS25	TV	B0000	1AS31				64206	16	30000	02037
0AS26	SN	3AS00	00036		A/7		64207	33	02061	00044
0AS27	RP	20007	1AS29		EVAL RAND		64210	75	20007	02035
0AS28	PM	3AS01	00025		POLY	29	64211	24	02062	00031
0AS29	MP	B0000	00T03			56	64212	71	30000	01325
0AS30	LA	A0000	00010			66	64213	54	20000	00012
0AS31	SP	B0000	00000				64214	31	30000	00000
0AS32	TP	B0000	00025			31	64215	11	30000	00031
0AS33	TP	00024	Q0000		X		64216	11	00030	10000

PX 71900-10-(149)

0AS34	QJ	1AS35	1AS37			X NEG		64217	44	02043	02045
0AS35	SP	3AS11	00001			PI		64220	31	02074	00001
0AS36	ST	00025	00025					64221	36	00031	00031
0AS37	IJ	00023	1AS40					64222	41	00027	02050
0AS38	SP	3AS11	00000					64223	31	02074	00000
0AS39	ST	00025	00025					64224	36	00031	00031
0AS40	SF	00025	00023					64225	74	00031	00027
0AS41	ZJ	1AS42	1AS01					64226	47	02052	02001
0AS42	LA	A0000	00027					64227	54	20000	00033
0AS43	TP	B0000	00025			M FINAL		64230	11	30000	00031
0AS44	SP	00023	00027					64231	31	00027	00033
0AS45	AT	3AS10	Q0000			E FINAL		64232	35	02073	10000
0AS46	CC	00025	Q0000			PACK		64233	27	00031	10000
0AS47	TP	A0000	A0000			EXTEND		64234	11	20000	20000
0AS48	MJ	00000	1AS01			OUT		64235	45	00000	02001
2AS00	01	26249	11000	-03	42	A/7		64236	05	12750	53762
2AS01	06	67009	01000	-03	40	A/6		64237	06	65103	05327
2AS02	-1	70881	25600	-02	38	A/5		64240	73	50016	32330
2AS03	03	08918	81000	-02	36	A/4		64241	01	76420	76052
2AS04	-5	01743	04600	-02	34	A/3		64242	77	14476	15552
2AS05	08	89789	87400	-02	32	A/2		64243	00	26616	51661
2AS06	-2	14598	80160	-01	30	A/1		64244	77	62210	01542
2AS07	01	57079	63050		28	A 0		64245	00	31103	75516
2AS08	06	20000	00000	B		50		64246	06	20000	00000
2AS09	10	00000	00000	B		1	33	64247	10	00000	00000
2AS10	07	40000	00000	B		60		64250	07	40000	00000
2AS11	01	57079	63268		31	PI OVER 2		64251	03	11037	55242
START								00000	45	00000	00000

PX 71900-10-(149)

THE RAMO-WOOLDRIDGE CORPORATION
Los Angeles 45, California

Floating Point Arctangent Routine

Specifications

Identification Tag:	TNI-1
Type:	Subroutine
Assembly Routine Spec:	SUB 50137 05114
Storage:	51 words total program storage.
	3 words temporary storage pool used, addresses 00027b thru 00031b
	The constant pool is used.
Entrance and Exit:	RJ 00K01 00K02
Drum Assignment:	Address 63631b thru 63713b
Machine Time:	4.27 ms average, 5.8 ms maximum
Mode of Operation:	Floating point

Coded by:	M. Perry	July, 1955
Code Checked by:	R. Bigelow	July, 1955
Machine Checked by:	M. Perry	August, 1955
Approved by:	W. Bauer	August, 1955

Description

When supplied with an argument X in SNAP form, this routine will evaluate Arctan X (in radians) using a Rand Polynomial Approximation producing an answer in SNAP form.

Programming Instructions

1. Place the double length extension of X in the accumulator.

   X must be in SNAP form.

2. Return Jump to the subroutine. Assuming that the subroutine was assigned to region 00K00 for assembly, use the instruction RJ 00K01 00K02.

3. At the time of exit from the subroutine, the double length extension of Arctan X will be in the accumulator in SNAP form. The range of the result will be the principal value, defined as follows:

$$-\pi/2 \leq \text{Arctan } X \leq \pi/2$$

Error Analysis

The error in the result produced by the routine is less than 2^{-25}.

Mathematical Analysis

1. If $x \geq 1$, the identity Arctan X = $(\pi/2)$ - Arctan (1/X) is used.

2. Rand Polynomial Number 13 is evaluated using X (or 1/X) as the argument.

Range of Variable

No alarm conditions exist. Any number which can be expressed in SNAP form can be entered and the result will have the accuracy stated above.

PX 71900-10-(74)

```
D            00T00 50137      ARCTN RTN 37   63631   00 00000 00000
D            01T00 01024      TO BE ALTERD   02000   00 00000 00000
D            02T00 50174      ARCTN CNS 14   63676   00 00000 00000
D            03T00 01061      TO BE ALTERD   02045   00 00000 00000
00T00   00   00000 00000      ALARM EXIT     63631   00 00000 00000
00T01   MJ   00000 00000      NORMAL EXIT    63632   45 00000 00000
00T02   TP   00013 00025      NORMAL ENTRY   63633   11 00015 00031
00T03   TP   03T00 01T35                     63634   11 02045 02043
00T04   SJ   01T05 01T06                     63635   46 02005 02006
00T05   TP   03T01 01T35      NEG ARG        63636   11 02046 02043
00T06   TM   A0000 Q0000                     63637   12 20000 10000
00T07   QT   03T02 00023      M              63640   51 02047 00027
00T08   SS   Q0000 00000                     63641   34 10000 00000
00T09   SA   03T03 00008      MINUS E        63642   32 02050 00010
00T10   TM   B0000 00024      E ABS          63643   12 30000 00030
00T11   SJ   01T12 01T15                     63644   46 02014 02017
00T12   SN   00015 00039      1              63645   33 00017 00047
00T13   DV   00023 00023      1 OVER M  28   63646   73 00027 00027
00T14   TP   03T04 00025      PI OVER 2      63647   11 02051 00031
00T15   SP   03T05 00000                     63650   31 02052 00000
00T16   SS   00024 00000                     63651   34 00030 00000
00T17   SJ   01T20 01T18                     63652   46 02024 02022
00T18   TV   A0000 01T19      42/E ABS       63653   16 20000 02023
00T19   LA   00023 00000                     63654   54 00027 00000
00T20   TP   B0000 00023      ARG OF POLY    63655   11 30000 00027
00T21   MP   B0000 Q0000                     63656   71 30000 10000
00T22   LA   A0000 00001                     63657   54 20000 00001
00T23   TP   B0000 00024      ARG SQUARED    63660   11 30000 00030
00T24   SN   03T06 00035      C/15           63661   33 02053 00043
00T25   RP   20007 01T27                     63662   75 20007 02033
00T26   PM   03T07 00024      POLY           63663   24 02054 00030
00T27   PM   00025 00023        EVALUATION   63664   24 00031 00027
00T28   TP   B0000 A0000                     63665   11 30000 20000
00T29   ZJ   01T30 01T01                     63666   47 02036 02001
00T30   SF   A0000 00024      SCALE          63667   74 20000 00030
00T31   TP   A0000 00023      M FINAL        63670   11 20000 00027
00T32   SP   00024 00000                     63671   31 00030 00000
00T33   SA   00014 00035      E/128 FINAL    63672   32 00016 00043
00T34   SA   00023 00027      PACK           63673   32 00027 00033
00T35   TP   B0000 A0000      EXTEND         63674   11 30000 20000
00T36   MJ   00000 01T01      OUT            63675   45 00000 02001
```

02T00	TP B0000 A0000				POS CONDITN	63676	11	30000	20000
02T01	TN B0000 A0000				NEG CONDITN	63677	13	30000	20000
02T02	00 07777 77777	B			MASK	63700	00	07777	77777
02T03	20 00000 00000	B			128	63701	20	00000	00000
02T04	01 57079 63268			27	PI OVER 2	63702	00	14441	76652
02T05	00 00000 00052	B			42	63703	00	00000	00052
02T06	04 05405 80000		-03	36	C/15	63704	00	20465	76350
02T07	02 18612 28800		-02	34	C/13	63705	00	26305	45073
02T08	-5 59098 86100		-02	33	C/11	63706	77	43277	43606
02T09	09 64200 44100		-02	32	C/9	63707	00	30535	75750
02T10	-1 39085 33510		-01	31	C/7	63710	77	56144	71644
02T11	01 99465 35990		-01	30	C/5	63711	00	14610	05133
02T12	-3 33298 56050		-01	29	C/3	63712	77	65253	17101
02T13	09 99999 33290		-01	28	C/1	63713	00	17777	77515
START									

PX 71900-10-(74)

THE RAMO-WOOLDRIDGE CORPORATION
Los Angeles 45, California

CHANGED WORD POST-MORTEM ROUTINE (revised)

Specifications

Identification Tag:	MDP-3
Type:	Service Toutine (with subroutine entrance)
Special Storage:	The constant pool and temporary storage pool are not used by this routine.
Service Entrance:	Address 40037b
Program Entrance:	40037b
Program Exit:	40020b
Alarm Exit:	The alarm exit is not used by this routine.
Machine Time:	(14.1 + .5n) seconds where n=number of cards punched.

Coded by:	R. Beach	October 26, 1955
Code Checked by:	R. Beach	October 26, 1955
Machine Checked by:	R. Beach	October 26, 1955
Revised by:	C. Koos	December 1, 1955
Approved by:	W. F. Bauer	December 9, 1955

PX 71900-10-(102)

Description

This routine compares ES with the MD image of ES and prints out those words of ES which are not the same as their correspondent in the image. ES is not altered by the routine, and the MD image is up-dated to be identical with ES when exit is made from the routine.

The routine stores ES at addresses 66000b to 67777b and reads portions of this image and the regular image (76000b - 77777b) into ES and compares words.

If the corresponding words are the same, they are replaced by zero, unless the new value is zero. In the latter case the word is replaced by 45 40037 40020b. The changed words and zeros are then read into ES. ES is then dumped on the line printer. (Note: Until the line printer is in use, this dump will be made onto cards by employing MDP-4). ES and the 76000b image are then restored from the 66000b image.

Each card contains six words. If any one word is zero, it should be ignored as it is not a changed word. A word which has been changed to zero has been given the arbitrary tag 45 40037 40020b and will be punched as such. Also, a word that was changed to this tag will be identified in the same manner. The programmer must therefore distinguish between these two cases.

Operating Instructions

1. When routine is used as a service routine set PAK to 40037b. Routine will find changed words, print them out, and stop on 56 00000 40037b.

2. When routine is used as a subroutine enter routine with 37 40020 40037b. Operation of routine is the same except that routine exits to address y+1 if y is the address of the RJ instruction used to enter the routine.

3. Most service routines use all or parts of ES and their activation will destroy the old 76000b image. Hence, if a changed word comparison is desired, the execution of MDP-3 must precede the use of other post-mortem routines.

Alarm Conditions

There are no alarm conditions in this routine. However, if the routine hangs up during punching, or if the machine is halted during punching, a start at 40040b will clear the punch, restore ES, and up-date the 76000b image.

PX 71900-10-(102)

THE RAMO-WOOLDRIDGE CORPORATION
Los Angeles 45, California

Nth Root Routine

Specifications

Identification Tag:	NRT-0
Type:	Subroutine
Assembly Routine Spec:	SUB 51316 03701
Storage:	36 instructions, addresses 10F00 thru 10F35 1 constant in program, address 10F36 37 words total program storage, addresses 10F00 thru 10F36 4 words temporary storage pool used, addresses 00027b thru 00033b The constant pool is used by this routine.
Program Entrance:	Address 10F02
Program Exit:	Address 10F01
Alarm Exit:	The alarm exit is used by this routine.
Drum Assignment:	Address 66064b thru 66130b
Machine Time:	Average execution time $2(n-2) + 5$ milliseconds for $n \leq 50$
Mode of Operation:	Fixed point

Coded by:	W. Frank	November 25, 1955
Code Checked by:	W. Frank	November 28, 1955
Machine Checked by:	W. Frank	November 30, 1955
Approved by:	W. F. Bauer	December 1, 1955

Description

This subroutine extracts the n^{th} root of any number M, scaled at 2^{35}, and such that

$$\left| M \cdot 2^{35} \right| \leq 2^{35} - 1.$$

n must be an integer in the range

$$2 \leq n < 2^{12}$$

The routine must be entered with $M \cdot 2^{35}$ in A and $n \cdot 2^{0}$ in Q. The result will be left in A, scaled at 2^{35}, at the conclusion of the routine.

Programming Instructions

1. Place $M \cdot 2^{35}$ in A_R. (A_L) is ignored by this routine.
2. Place $n \cdot 2^{0}$ in Q.
3. Enter the subroutine with RJ 00F01 00F02, where 00F00 is the location of the first word of NRT-0.
4. The subroutine returns control to the cell following the RJ instruction with $(\sqrt[n]{M}) \cdot 2^{35}$ in A.

Alarm Conditions

The subroutine enters the alarm routine ALR-1 if n is negative or M is negative for n even. In either case, the word "alarm" is printed on the flexowriter, followed by the octal address of the RJ instruction used to enter NRT-0.

Execution Time

The time taken to find the n^{th} root of a number is inversely proportional to the magnitude of the number and directly proportional to the size of n. An average estimate, for $n \leq 50$, is approximately $3(n-2) + 5$ milliseconds.

PX 71900-10-(116)

Mathematical Method

An iterative procedure, employing the Newton-Raphson method#, is used to solve the equation

$$x^n = M$$

The process is of second order and is defined by

$$x_{i+1} = x_i + \frac{1}{n}\left[\frac{|M|}{(x_i)^{n-1}} - x_i\right]$$

where $x_0 \cdot 2^{35} = 2^{35} - 1$

The iteration is terminated when

$$\frac{|M|}{(x_i)^{n-1}} - x_i \geq 0$$

A secondary test is made to insure

$$|M| \leq (x_i)^{n-1}$$

This test is necessary, eventhough the process is monotonic; for, it is possible that truncation of the result of multiplication and division can violate this property. In that event, x_{i-1} is taken as the solution. A special case is M = 0, where the solution is x = 0 for all p.

Accuracy

The error in the result of this routine was found to be less than 10^{-10}; that is, for an input argument, which is correct to 35 bits, one can expect an answer which may be incorrect at most in the right octal digit.

#Scarborough, J.B., Numerical Mathematical Analysis, second edition, The John Hopkins Press, Baltimore, Md., 1950, p. 192.

```
D              00F00  01024                                2000  00  00000  00000
D              10F00  51316                               66064  00  00000  00000
D              0CP00  00013                                  15  00  00000  00000
10F00    37    75701  75702  B       ALARM                66064  37  75701  75702
10F01    MJ    00000  0              EXIT                 66065  45  00000  00000
10F02    ZJ    00F03  00F01          ARG ZERO             66066  47  02003  02001
10F03    TP    A0000  0CP11                               66067  11  20000  00030
10F04    TP    Q0000  0CP10                               66070  11  10000  00027
10F05    QT    00016  A0000          N EVEN OR            66071  51  00020  20000
10F06    ZJ    00F09  00F07            OD                 66072  47  02011  02007
10F07    TP    0CP11  A0000                               66073  11  00030  20000
10F08    SJ    00F00  00F09          ALARM                66074  46  02000  02011
10F09    TM    0CP11  0CP14                               66075  12  00030  00033
10F10    TP    00F36  0CP12          SET X0 VALUE         66076  11  02044  00031
10F11    54    0CP10  20017  BRB                          66077  54  00027  20017
10F12    SJ    00F00  00F13          N NEGATIVE           66100  46  02000  02015
10F13    TU    A0000  00F17                               66101  15  20000  02021
10F14    RS    00F17  00015                               66102  23  02021  00017
10F15    RS    00F17  00015                               66103  23  02021  00017
10F16    SP    0CP12  00035          SET UP B             66104  31  00031  00043
10F17    RP    00000  00F19                               66105  75  00000  02023
10F18    MP    B0000  0CP12          XITH TO N-1          66106  71  30000  00031
10F19    TP    B0000  0CP13                               66107  11  30000  00032
10F20    TP    0CP14  A0000                               66110  11  00033  20000
10F21    TJ    0CP13  00F23                               66111  42  00032  02027
10F22    MJ    00000  00F30                               66112  45  00000  02036
10F23    LA    A0000  00035                               66113  54  20000  00043
10F24    DV    0CP13  A0000                               66114  73  00032  20000
10F25    ST    0CP12  A0000                               66115  36  00031  20000
10F26    SJ    00F27  00F30          CONVFRGENCE          66116  46  02033  02036
10F27    DV    0CP10  A0000                               66117  73  00027  20000
10F28    AT    0CP12  0CP12          XITH PLUS 1          66120  35  00031  00031
10F29    MJ    00000  00F16                               66121  45  00000  02020
10F30    TP    0CP11  A0000                               66122  11  00030  20000
10F31    SJ    00F32  00F34                               66123  46  02040  02042
10F32    TN    0CP12  A0000                               66124  13  00031  20000
10F33    MJ    00000  00F01                               66125  45  00000  02001
10F34    TP    0CP12  A0000                               66126  11  00031  20000
10F35    MJ    00000  00F01                               66127  45  00000  02001
10F36    37    77777  77777  B                            66130  37  77777  77777
```

PX 71900-10-(116)

THE RAMO-WOOLDRIDGE CORPORATION
Los Angeles 45, California

Gill Method Subroutine

Specifications

Identification Tag:	NUI-3
Type:	Subroutine
Assembly Storage Spec:	SUB 49880 07414
Storage:	59 instructions, addresses 0GM00 thru 0GM40 1GM00 thru 1GM17 15 constants in program, addresses 0GC00 thru 0GC14 74 words total program storage, addresses 0GM00 thru 0GM40 1GM00 thru 1GM17 0GC00 thru 0GC14 10 words temporary storage pool used, addresses 00027b (0GT00) thru 00040b (0GT09)
Drum Assignment:	Addresses 63230b thru 63341b
Program Entrances:	Addresses 0GM02, 0GM03, and 0GM04
Program Exit:	Address 0GM01
Machine Time:	(10.3 n + 1.9) ms per point average, where n equals the number of equations in the system
Mode of Operation:	Fixed point
Coded by:	J. Carlson R. Douthitt M. Elmore R. Summers
Code Checked by:	M. Elmore — June 8, 1955
Machine Checked by:	M. Elmore — July 7, 1955
Approved by:	W. Bauer — July 22, 1955

Description

The Gill Method Subroutine integrates a system of first order, differential equations using a step-by-step process. Using the values of the variables at a point and the coding for computing the derivative of each of the dependent variables at that point, the Gill Method Subroutine produces the coordinates for the next point of the solution each time it is entered.

A special entrance sets up the subroutine for a particular system of equations, thus allowing the subroutine to solve concurrently several different systems in the same program.

The independent variable is incremented within the subroutine itself.

Notation

The system of equations to be solved is

$$\frac{dy_i}{dx} = f_i\,(x, y_1, y_2, \ldots, y_n),\ (i = 1, 2, \ldots, n).$$

q_i are intermediate values of the calculation (zero initially)

Δx is the increment of the independent variable x

h is the binary scaling power of x (i.e. $x \cdot 2^h$ is in the computer)

h-1 is the binary scaling power of Δx

m_i is the binary scaling power of y_i

f is the common difference between the scaling power of y_i and the scaling power of $\frac{dy_i}{dx}$ for each i.

$m_i - f$ is the binary scaling power of $\frac{dy_i}{dx}$

$L = 73 + f - h$

Programming and Operating Instructions

Assign the Gill Method Subroutine to some arbitrary region, say OGMOO.

In order to solve a given system, the following array of variables, derivatives, intermediate values, and parameters should be assigned a region, say OGNOO.

PX 71900-10-(91)

OGN00		L	
OGN01	00 OGN05 OGN06		
OGN02		n-1	
OGN03		Δx	scaled 2^{h-1}
OGN04		x	scaled 2^h
OGN05		$\frac{dy_1}{dx}$	scaled $m_1 - f$
OGN06		y_1	scaled m_1
OGN07		q_1	initially zero
OGN08		$\frac{dy_2}{dx}$	scaled $m_2 - f$
OGN09		y_2	scaled m_2
OGN10		q_2	initially zero

The q_i must be set to zero initially by the programmer.

In addition, the coding for computing $\frac{dy_i}{dx}$ for all i, (i = 1, 2, . . . , n) should be assigned a region, say ODE00. This coding will use the values in region OGN00 to compute all $\frac{dy_i}{dx}$ as specified by the equations in the system and should place the results in the appropriate places in region OGN00. It should then exit to the Gill Method Subroutine with an MJ 00000 OGM04 (see below).

Assuming the Gill Method Subroutine is in region OGM00, the three entrances are OGM02, OGM03, and OGM04. The exit is OGM01.

The first entrance, OGM02, is used for setting up the Gill Method Subroutine only for the particular system to be solved. It is entered by an RJ command followed by a parameter word which specifies the location of the variables, and the location of the coding for calculating the derivatives:

```
RJ OGM01 OGM02
00 OGN00 ODE00
```

The second entrance, OGM03, is the entrance for producing a point of the solution. It is entered by an RJ command: RJ OGM01 OGM03. Entering using this command results in four passes through both the Gill Method Subroutine and the coding for computing the derivatives, and leaves in region OGN00 the new values of the variables, the derivatives at those values, and x advanced by Δx, ready for the next step.

The third entrance, OGM04, is the entrance from the coding for calculating the derivatives and is used on each of the four passes necessary for computing one point. As noted above, it is entered by an MJ command in the ODE00 region:

Mathematical Analysis

Theory. "A Process for the Step-by-Step Integration of Differential Equations in an Automatic Digital Computing Machine" by S. Gill, published in Cambridge Philosophical Society Proceedings, Vol. 47, Part I, January 1951, should be consulted for a detailed analysis of the process on which the subroutine is based.

Suppose we know the point $(X, Y_1, Y_2, \ldots, Y_n)$ on the curve defined by the system of equations

$$\frac{dy_1}{dx} = f_1(x, y_1, y_2, \ldots, y_n)$$

$$\frac{dy_2}{dx} = f_2(x, y_1, y_2, \ldots, y_n)$$

$$\vdots$$

$$\frac{dy_n}{dx} = f_n(x, y_1, y_2, \ldots, y_n)$$

The Gill Method is a process by which we can find the next point on the curve: i.e. the value of $y_1, y_2, \ldots, y_n$ for $x = X + h$.

The process can be better understood if the case where n=1 is first considered.

We have the point (X,Y) on the curve $\frac{dy}{dx} = f(x,y)$, and we want to find y at X + h; i.e. we want $k = \delta y$ such that $\left.\frac{dy}{dx}\right]_{X+h,\, Y+k} = f(X + h, Y + k)$.

We derive k by making four approximations and averaging them in a particular way.

First approximate the curve by a straight line through (X,Y) with the slope $\left.\frac{dy}{dx}\right]_{X,Y} = f(X,Y)$, and find a first approximation to k:

$$k_o = h \cdot f(X,Y)$$

Then we travel a fraction m of the way along this line to the point $(X + mh, Y + mk_o)$ and find $f(X + mh, Y + mk_o)$.

This gives us a new straight line through $(X + mh, Y + mk_o)$ with slope $f(X + mh, Y + mk_o)$, and we find

$$k_1 = h\, f(X + mh, Y + mk_o)$$

We now use k_o and k_1 to find a third point at which f is calculated: $(X + nh, Y + [n-r]\, k_o + rk_1)$.

$$k_2 = h\, f(X + nh, Y + [n-r]\, k_o + rk_1)$$

PX 71900-10-(91)

Similarly,

$$k_3 = h \cdot f\ (X + ph,\ Y + \left[p-s-t\right]\ k_o + sk_1 + tk_2$$

The weighted average of k_o, k_1, k_2, and k_3 is the desired $k = \delta y$:

$$\delta y = y\ (X + h) - y\ (X) = c_o k_o + c_1 k_1 + c_2 k_2 + c_3 k_3$$

where $c_o + c_1 + c_2 + c_3 = 1$.

For a system of equations, the same four steps given above are made for each equation and

$$\delta y_i = c_o k_{io} + c_1 k_{i1} + c_2 k_{i2} + c_3 k_{i3} \text{ where } c_o + c_1 + c_2 + c_3 = 1.$$

The above process is, for certain values of m, n, p, s, t, c_o, c_1, c_2, and c_3, the Runge-Kutta process. The Gill process was derived, with application to machine use in mind, by minimizing the number of storage cells required. For the Gill Method the above constants are

$m = 1/2$,	$r = 1 - \sqrt{1/2}$,	$c_o = 1/6$
$n = 1/2$,	$s = - \sqrt{1/2}$,	$c_1 = (1/3)\ (1 - \sqrt{1/2})$
$p = 1$,	$t = 1 + \sqrt{1/2}$,	$c_2 = (1/3)\ (1 + \sqrt{1/2})$
		$c_3 = 1/6$

The Gill process further systematizes the calculation so as to increase the accuracy and simplify the coding.

<u>The Subroutine</u> As used in the Gill Method Subroutine, the process is as follows:

1st pass:

Advance x by $(1/2)h$

$$k_{1o} = h \cdot f_1\ (x, y_{10}, y_{20}, \ldots, y_{no})$$

$$r_{11} = (1/2)k_{1o} - q_{1o}$$

$$q_{11} = q_{1o} + 3r_{11} - (1/2)k_{1o}$$

$$y_{11} = y_{1o} + r_{11}$$

Calculate $f_1\ (x, y_{11}, y_{21}, \ldots, y_{n1})$ in programmer's own coding.

2nd pass:

$$k_{11} = h\ f_1\ (x, y_{11}, y_{21}, \ldots, y_{n1})$$

$$r_{12} = (1 - \sqrt{1/2})\ (k_{11} - q_{11})$$

$$q_{12} = q_{11} + 3r_{12} - (1 - \sqrt{1/2})\ k_{11}$$

$$y_{12} = y_{11} + r_{12}$$

Calculate $f_1\ (x, y_{12}, y_{22}, \ldots, y_{n2})$ in programmer's own coding.

3rd pass:

Advance x by (1/2)h

$k_{i2} = h \cdot f_i \, (x, y_{12}, y_{22}, \ldots, y_{n2})$

$r_{i3} = (1 + \sqrt{1/2})\,(k_{i2} - q_{i2})$

$q_{i3} = q_{i2} + 3r_{i3} - (1 + \sqrt{1/2})\,k_{i2}$

$y_{i3} = y_{i2} + r_{i3}$

Calculate $f_i \, (x, y_{13}, y_{23}, \ldots, y_{n3})$ in programmer's own coding.

4th pass:

$k_{i3} = h \cdot f_i \, (x, y_{13}, y_{23}, \ldots y_{n3})$

$r_{i4} = (1/6)(k_{i3} - 2q_{i3})$

$q_{i4} = q_{i3} - 3r_{i4} - (1/2)k_{i3}$

$y_{i4} = y_{i3} + r_{i4}$

Calculate $f_i(x, y_{14}, y_{24}, \ldots, y_{n4})$ in programmer's own coding.

Machine Checking

A driver routine solved two systems of equations both separately and concurrently, using the Gill Method Subroutine. The two systems solved are given below to indicate accuracy and to serve as examples.

PX 71900-10-(91)

1. Equations

$$\left.\begin{aligned} \frac{dy_1}{dx} &= y_2 \\ \frac{dy_2}{dx} &= -y_1 \end{aligned}\right\}$$ equivalent to the second order equation,

$$\frac{d^2y}{dx^2} + y = 0.$$

$\Delta x = .0872664626 = \pi/36 = 5^\circ$

Initial Conditions

At $x=0$, $y_1 = 0$ and $y_2 = 1$.

Solution

$y_1 = \sin x$

Accuracy

In a spot check of the results, the greatest absolute error observed was 1.5×10^{-6}. (For $x = 3.1415925696$, $y_1 = .0000015425$. However, $\sin x = .000000084$).

2. Equations

$$\left.\begin{aligned} \frac{dy_1}{dx} &= y_2 \\ \frac{dy_2}{dx} &= y_3 \\ \frac{dy_3}{dx} &= \frac{y_3 + 4x^2}{x} \end{aligned}\right\}$$ Equivalent to the third order equation

$$x\frac{d^3y}{dx^3} - \frac{d^2y}{dx^2} = 4x^2$$

$\Delta x = .1$

Initial Conditions

At $x = .1$, $y_1 = .000025$, $y_2 = .001$, $y_3 = .03$

Solution

$$y_1 = \frac{x^4}{3} - \frac{x^3}{60} + \frac{x}{6000} - \frac{1}{120,000}$$

Accuracy

In a spot check of the results, the greatest relative error observed was 3.4×10^{-6}. (For $x - .1999999975$, $y_1 = .00042499858$. However, the solution is actually .00042500002).

```
D               0GM   [illegible]                             [illegible]  00 00000 00000
D               1GM   [illegible]                             02051  00 00000 00000
D               2GM   [illegible]                             63230  00 00000 00000
D               3GM   49921                                   63301  00 00000 00000
D               0GC   1083                                    02073  00 00000 00000
D               1GC   49939                                   63323  00 00000 00000
D               0GT      23                                   00027  00 00000 00000
2GM00   00 00000 00000                                        63230  00 00000 00000
2GM01   MJ                                                    63231  45 00000 00000
2GM02   MJ             1GM00        TO SET UP                 63232  45 00000 02051
2GM03   MJ             0GM36        ENTER                     63233  45 00000 02044
2GM04   TP 0GM00 Q0000              RE ENTER                  63234  11 02000 10000
2GM05   QJ 0GM06 0GM40              WHAT                      63235  44 02006 02050
2GM06   QJ 0GM38 0GM07                PASS                    63236  44 02046 02007
2GM07   TP Q0000 0GM00              STORE CTR                 63237  11 10000 02000
2GM08   RA 0GM10 0GC01              UP PASS                   63240  21 02012 02074
2GM09   RP 30003 0GM11              STORE                     63241  75 30003 02013
2GM10   TP       0GT05                PASS CONS               63242  11 00000 00034
2GM11   TU       0GM15              RESET                     63243  15 00000 02017
2GM12   TV       0GM31                ADDRESSES               63244  16 00000 02037
2GM13   TP       0GT08              RESET N 1                 63245  11 00000 00037
2GM14   RP 30003 0GM16                                        63246  75 30003 02020
2GM15   TP       0GT00                                        63247  11 00000 00027
2GM16   MP       0GT00              CALC K                    63250  71 00000 00027
2GM17   LA A                          SHIFT L                 63251  54 20000 00000
2GM18   TP A     0GT03              STORE K                   63252  11 20000 00032
2GM19   MP 0GT05 0GT03              AK                        63253  71 00034 00032
2GM20   MA 0GT06 0GT02              AK BQ                     63254  72 00035 00031
2GM21   LA A        38                                        63255  54 20000 00046
2GM22   TP A     0GT04              STORE R                   63256  11 20000 00033
2GM23   AT A     A                  2R                        63257  35 20000 20000
2GM24   AT 0GT04 A                  3R                        63260  35 00033 20000
2GM25   LA A        34                                        63261  54 20000 00042
2GM26   MA 0GT07 0GT03              3R CK                     63262  72 00036 00032
2GM27   LA A        38              ADD OLD Q                 63263  54 20000 00046
2GM28   AT 0GT02 0GT02              EQUALS Q                  63264  35 00031 00031
2GM29   RA 0GT01 0GT04              NEW Y                     63265  21 00030 00033
2GM30   RP 30002 0GM32              STORE                     63266  75 30002 02040
2GM31   TP 0GT01                      Y AND Q                 63267  11 00030 00000
2GM32   RA 0GM15 0GC01              ADVANCE                   63270  21 02017 02074
2GM33   RA 0GM31 0GC02                ADDRESSES               63271  21 02037 02075
2GM34   IJ 0GT08 0GM14              CYCLE N                   63272  41 00037 02016
2GM35   MJ                                                    63273  45 00000 00000
2GM36   TU 0GC00 0GM10                                        63274  15 02073 02012
2GM37   TP 0GM21 Q                                            63275  11 02025 10000
2GM38   RA                                                    63276  21 00000 00000
2GM39   MJ             0GM07                                  63277  45 00000 02007
```

PX 71900-10-(91)

```
2GM40   QJ 0GM01 0GM07                                   63300  44 02001 02007
3GM00   SP 0GM01 00015                                   63301  31 02001 00017
3GM01   TU A     1GM02                                   63302  15 20000 02053
3GM02   TP       0GT00                                   63303  11 00000 00027
3GM03   TV 0GT00 0GM35                                   63304  16 00027 02043
3GM04   TU 0GT00 1GM05                                   63305  15 00027 02056
3GM05   TV       0GM17                                   63306  16 00000 02021
3GM06   RA 0GT00    15                                   63307  21 00027 00017
3GM07   RP 10003 1GM09                                   63310  75 10003 02062
3GM08   TU 0GT00 0GM11                                   63311  15 00027 02013
3GM09   RA 0GM13    15                                   63312  21 02015 00017
3GM10   AT    15 A                                       63313  35 00017 20000
3GM11   TU A     0GM16                                   63314  15 20000 02020
3GM12   TU A     0GM38                                   63315  15 20000 02046
3GM13   LQ A        21                                   63316  55 20000 00025
3GM14   TV Q     0GM38                                   63317  16 10000 02046
3GM15   RA 0GM38    15                                   63320  21 02046 00017
3GM16   RA 0GM01    16                                   63321  21 02001 00020
3GM17   MJ       0GM01                                   63322  45 00000 02001
1GC00      GC                                            63323  00 02073 00000
1GC01          3        B                                63324  00 00003 00000
1GC02                 3 B                                63325  00 00000 00003
1GC03    5              - 1   34  A1 SCALE 34            63326  10 00000 00000
1GC04   -1                    34  B1       34            63327  57 77777 77777
1GC05   -5              - 1   34  C1       34            63330  67 77777 77777
1GC06   02 92893 21881  - 1   34  A2 SCALE 34            63331  04 53730 31460
1GC07   -2 92893 21881  - 1   34  B2       34            63332  73 24047 46317
1GC08   -2 92893 21881  -01   34  C2 SCALE 34            63333  73 24047 46317
1GC09    1 70710 67812        34  A3       34            63334  33 24047 46320
1GC10   -1 70710 67812        34  B3       34            63335  44 53730 31457
1GC11   -1 70710 67812        34  C3       34            63336  44 53730 31457
1GC12    1 66666 66667  - 1   34  A4       34            63337  02 52525 25253
1GC13   -3 33333 33333  - 1   34  B4       34            63340  72 52525 25252
1GC14   -5              - 1   34  C4       34            63341  67 77777 77777
```

THE RAMO-WOOLDRIDGE CORPORATION
Los Angeles 45, California

Floating Point Gill Method

Identification Tag:	NUI-4		
Type:	Subroutine		
Assembly Routine Spec:	SUB 51921 08914		
Storage:	89 words total program storage		
	9 words temporary pool used, addresses 27 b thru 37 b		
	The constant pool is used.		
Entrance and Exit:	RJ GIL01 GIL02	set up	
	RJ GIL01 GIL03	to get next point	
	MJ 0 GIL04	From derivative calculation	
Machine Time:	Approximately (9.7 + 84.6n) m.s. per time interval, where n equals the number of equations in the system.		

Coded by:	J. Carlson	May, 1956
Approved by:	W. F. Bauer	May 10, 1956

PX 71900-10-(143)

Description

The Gill Method Subroutine integrates a system of first order, differential equations using a step-by-step process. Using the values of the variables at a point and the coding for computing the derivative of each of the dependent variables at that point, the Gill Method Subroutine produces the coordinates for the next point of the solution each time it is entered.

A special entrance sets up the subroutine for a particular system of equations, thus allowing the subroutine to solve concurrently several different systems in the same program.

The independent variable is incremented within the subroutine itself.

Notation

The system of equations to be solved is

$$\frac{dy_i}{dx} = f_i\,(x, y_1, y_2, \ldots, y_n), \; (i = 1, 2, \ldots, n).$$

n is the number of equations in the system.

q_i are intermediate values of the calculation (zero initially)

x is the increment of the independent variable x

Programming and Operating Instructions

SNAP must be in E.S.

Assign the Gill Method Subroutine to some arbitrary region, say GIL00. This region need not be located in the low numbered half of E.S.

In order to solve a given system, the following array of variables, derivatives, intermediate values, and parameters should be assigned a region, say DEQ00. Although the programmer will undoubtedly desire to have this region located in the low numbered half of E.S., it is not necessary for the operation of this subroutine.

DEQ00	n	Fixed point form scaled 2^0.
DEQ01	Δx	Floating point form
DEQ02	x	"
DEQ03	$\frac{dy_1}{dx}$	"
DEQ04	y_1	"
DEQ05	q_1	"

(The q_i must be set to zero initially by the programmer.)

DEQ06 $\frac{dy_2}{dx}$

DEQ07 y_2

DEQ08 q_2

In addition, the coding for computing $\frac{dy_i}{dx}$ for all i, (i = 1, 2, . . ., n) should be assigned a region, say DFQ00. This coding will use the values in region DEQ00 to compute all $\frac{dy_i}{dx}$ as specified by the equations in the system and should place the results in the appropriate places in region DEQ00. It should then exit to the Gill Method Subroutine with an MJ 00000 GIL04 (see below).

Assuming the Gill Method Subroutine is in region GIL00, the three entrances are GIL02, GIL03, and GIL04. The exit is GIL01.

The first entrance, GIL02, is used for setting up the Gill Method Subroutine only for the particular system to be solved. It is entered by an RJ command followed by a parameter word which specifies the location of the variables, and the location of the coding for calculating the derivatives:

```
RJ  GIL01  GIL02
00  DEQ00  DFQ00
```

The second entrance, GIL03, is the entrance for producing a point of the solution. It is entered by an RJ command: RJ GIL01 GIL03. Entering using this command results in four passes through both the Gill Method Subroutine and the coding for computing the derivatives, and leaves in region DEQ00 the new values of the variables, the derivatives at those values, and x advanced by Δx, ready for the next step.

The third entrance, GIL04, is the entrance from the coding for calculating the derivatives and is used on each of the four passes necessary for computing one point. As noted above, it is entered by an MJ command in the DFQ00 region:

```
MJ  00000  GIL04
```

Mathematical Analysis

Theory. "A Process for the Step-by-Step Integration of Differential Equations in an Automatic Digital Computing Machine" by S. Gill, published in Cambridge Philosophical Society Proceedings, Vol. 47, Part I, January 1951, should be consulted for a detailed analysis of the process on which the subroutine is based.

Suppose we know the point $(X, Y_1, Y_2, \ldots, Y_n)$ on the curve defined by the system of equations

$$\frac{dy_1}{dx} = f_1(x, y_1, y_2, \ldots, y_n)$$

PX 71900-10-(143)

$$\frac{dy_2}{dx} = f_2\,(x, y_1, y_2, \ldots, y_n)$$

$$\vdots$$

$$\frac{dy_n}{dx} = f_n\,(x, y_1, y_2, \ldots, y_n)$$

The Gill Method is a process by which we can find the next point on the curve: i.e. the value of $y_1, y_2, \ldots, y_n$ for $x = X + h$.

The process can be better understood if the case where $n = 1$ is first considered.

We have the point (X,Y) on the curve $\frac{dy}{dx} = f(x,y)$, and we want to find y at $X + h$; i.e. we want $k = \delta y$ such that $\left.\frac{dy}{dx}\right]_{X+h,\,Y+k} = f(X + h, Y + k)$.

We derive k by making four approximations and averaging them in a particular way.

First approximate the curve by a straight line through (X,Y) with the slope $\left.\frac{dy}{dx}\right]_{X,Y} = f(X,Y)$, and find a first approximation to k:

$$k_o = h \cdot f(X,Y)$$

Then we travel a fraction m of the way along this line to the point $(X+mh, Y+mk_o)$ and find $f(X + mh, Y + mk_o)$.

This gives us a new straight line through $(X + mh, Y + mk_o)$ with slope $f(X + mh, Y + mk_o)$, and we find

$$k_1 = h\, f(X + mh, Y + mk_o)$$

We now use k_o and k_1 to find a third point at which f is calculated: $(X + nh, Y + [n-r]\, k_o + rk_1)$.

$$k_2 = h\, f(X + nh, Y + [n-r]\, k_o + rk_1)$$

Similarly,

$$k_3 = h \cdot f(X + ph, Y + [p-s-t]\, k_o + sk_1 + tk_2$$

The weighted average of k_o, k_1, k_2, and k_3 is the desired $k = \delta y$:

$$\delta y = y(X + h) - y(X) = c_o k_o + c_1 k_1 + c_2 k_2 + c_3 k_3$$

where $c_o + c_1 + c_2 + c_3 = 1$.

For a system of equations, the same four steps given above are made for each equation and

$$\delta y_i = c_o k_{io} + c_1 k_{i1} + c_2 k_{i2} + c_3 k_{i3} \text{ where } c_o + c_1 + c_2 + c_3 = 1.$$

The above process is, for certain values of m, n, p, s, t, c_o, c_1, c_2, and c_3, the Runge-Kutta process. The Gill process was derived, with application to machine use in mind, by minimizing the number of storage cells required. For the Gill Method the above constants are

$m = 1/2$, $r = 1 - \sqrt{1/2}$, $c_o = 1/6$

$n = 1/2$, $s = -\sqrt{1/2}$, $c_1 = (1/3)(1 - \sqrt{1/2})$

$p = 1$, $t = 1 + \sqrt{1/2}$, $c_2 = (1/3)(1 + \sqrt{1/2})$

$c_3 = 1/6$

The Gill process further systematizes the calculation so as to increase the accuracy and simplify the coding.

The Subroutine. As used in the Gill Method Subroutine, the process is as follows:

It is assumed that the $f_i(x, y_{10}, y_{20}, \ldots, y_{no})$ and the y_i are available.

1st pass:

Advance x by (1/2)h

$k_{io} = h \cdot f_i (x, y_{10}, y_{20}, \ldots, y_{no})$

$r_{i1} = (1/2)k_{io} - q_{io}$

$q_{i1} = q_{io} + 3r_{i1} - (1/2)k_{io}$

$y_{i1} = y_{io} + r_{i1}$

Calculate $f_i (x, y_{11}, y_{21}, \ldots, y_{n1})$ in programmer's own coding.

2nd pass:

$k_{i1} = h f_i (x, y_{11}, y_{21}, \ldots, y_{n1})$

$r_{i2} = (1 - \sqrt{1/2})(k_{i1} - q_{i1})$

$q_{i2} = q_{i1} + 3r_{i2} - (1 - \sqrt{1/2})k_i$

$y_{i2} = y_{i1} + r_{i2}$

Calculate $f_i (x, y_{12}, y_{22}, \ldots, y_{n2})$ in programmer's own coding.

PX 71900-10-(143)

3rd pass:

Advance x by (1/2)h

$k_{i2} = h \cdot f_i \ (x, y_{12}, y_{22}, \ldots, y_{n2})$

$r_{i3} = (1 + \sqrt{1/2}) \ (k_{i2} - q_{i2})$

$q_{i3} = q_{i2} + 3r_{i3} - (1 + \sqrt{1/2}) \ k_{i2}$

$y_{i3} = y_{i2} + r_{i3}$

Calculate $f_i \ (x, y_{13}, y_{23}, \ldots, y_{n3})$

4th pass:

$k_{i3} = h \cdot f_i \ (x, y_{13}, y_{23}, \ldots y_{n3})$

$r_{i4} = (1/6)(k_{i3} - 2q_{i3})$

$q_{i4} = q_{i3} - 3r_{i4} - (1/2)k_{i3}$

$y_{i4} = y_{i3} + r_{i4}$

Calculate $f_i \ (x, y_{14}, y_{24}, \ldots, y_{n4})$ in programmer's own coding.

Machine Checking

The following system of two equations was solved using this routine:

$$\frac{dy_1}{dx} = \cos x$$

$$\frac{dy_2}{dx} = -\sin x$$

The initial conditions, at $x = 0$, were

$$y_1 = 0 \quad \text{and} \quad y_2 = 1$$

The interval, Δx, used was $2\pi/360$ radians. At $x = 360^\circ$ the results were accurate to 8 decimal digits.

D		GIL00	01024	02000	00	00000	00000
D		GIM00	51921	67221	00	00000	00000
GIM00	MS	00000	GIL00	67221	56	00000	02000
GIM01	MJ	00000	00000	67222	45	00000	00000
GIM02	MJ	00000	GIL49	67223	45	00000	02061
GIM03	MJ	00000	GIL08	67224	45	00000	02010
GIM04	RA	GIL71	00016	67225	21	02107	00020
GIM05	EJ	GIL69	GIL73	67226	43	02105	02111
GIM06	EJ	GIL68	GIL01	67227	43	02104	02001
GIM07	MJ	00000	GIL18	67230	45	00000	02022
GIM08	TP	00016	GIL71	67231	11	00020	02107
GIM09	TU	GIL72	GIL20	67232	15	02110	02024
GIM10	TP	00000	GIL88	67233	11	00000	02130
GIM11	TP	00000	00024	67234	11	00000	00030
GIM12	TP	GIL88	00002	67235	11	02130	00002
GIM13	ADNO	00024	00000	67236	14	04030	00000
GIM14	TP	00002	GIL87	67237	11	00002	02127
GIM15	TP	GIL75	00031	67240	11	02113	00037
GIM16	ADMP	00024	00031	67241	14	04030	14037
GIM17	TP	00002	00000	67242	11	00002	00000
GIM18	TV	00000	GIL70	67243	16	00000	02106
GIM19	RP	30003	GIL21	67244	75	30003	02025
GIM20	TP	00000	00023	67245	11	00000	00027
GIM21	RA	GIL20	GIL67	67246	21	02024	02103
GIM22	TU	GIL66	GIL25	67247	15	02102	02031
GIM23	TV	GIL66	GIL44	67250	16	02102	02054
GIM24	RP	30003	GIL26	67251	75	30003	02032
GIM25	TP	00000	00026	67252	11	00000	00032
GIM26	TP	00024	00002	67253	11	00030	00002
GIM27	MPNO	00028	00000	67254	14	14034	00000
GIM28	TP	00002	00029	67255	11	00002	00035
GIM29	TP	GIL88	00002	67256	11	02130	00002
GIM30	MPNO	00026	00000	67257	14	14032	00000
GIM31	TP	00002	00030	67260	11	00002	00036
GIM32	MPSU	00023	00029	67261	14	14027	10035
GIM33	TP	00002	00029	67262	11	00002	00035
GIM34	ADNO	00027	00000	67263	14	04033	00000
GIM35	TP	00002	00027	67264	11	00002	00033
GIM36	TN	00025	00002	67265	13	00031	00002
GIM37	MPAD	00030	00028	67266	14	14036	04034
GIM38	TP	00002	00030	67267	11	00002	00036
GIM39	TP	GIL85	00031	67270	11	02125	00037
GIM40	TP	00029	00002	67271	11	00035	00002
GIM41	DVAD	00031	00030	67272	14	20037	04036
GIM42	TP	00002	00028	67273	11	00002	00034

```
GIM43   RP 30003 GIL45                    67274   75 30003 02055
GIM44   TP 00026 00000                    67275   11 00032 00000
GIM45   RA GIL25 GIL67                    67276   21 02031 02103
GIM46   RA GIL44 GIL69                    67277   21 02054 02105
GIM47   RS GIL70 00016                    67300   23 02106 00020
GIM48   ZJ GIL24 00000                    67301   47 02030 00000
GIM49   TP GIL01 A0000                    67302   11 02001 20000
GIM50   LA A0000 00015                    67303   54 20000 00017
GIM51   TU A0000 GIL52                    67304   15 20000 02064
GIM52   TP 00000 A0000                    67305   11 00000 20000
GIM53   TV A0000 GIL48                    67306   16 20000 02060
GIM54   TU A0000 GIL18                    67307   15 20000 02022
GIM55   AT 00015 A0000                    67310   35 00017 20000
GIM56   TU A0000 GIL10                    67311   15 20000 02012
GIM57   AT 00015 A0000                    67312   35 00017 20000
GIM58   TU A0000 GIL11                    67313   15 20000 02013
GIM59   TU A0000 GIL66                    67314   15 20000 02102
GIM60   LA A0000 00057                    67315   54 20000 00071
GIM61   TV A0000 GIL17                    67316   16 20000 02021
GIM62   TV A0000 GIL66                    67317   16 20000 02102
GIM63   RA GIL66 00017                    67320   21 02102 00021
GIM64   RA GIL01 00016                    67321   21 02001 00020
GIM65   MJ 00000 GIL01                    67322   45 00000 02001
GIM66   00 00000 00000 B                  67323   00 00000 00000
GIM67   00 00003 00000 B                  67324   00 00003 00000
GIM68   00 00000 00005 B                  67325   00 00000 00005
GIM69   00 00000 00003 B                  67326   00 00000 00003
GIM70   00 00000 00000 B                  67327   00 00000 00000
GIM71   00 00000 00000 B                  67330   00 00000 00000
GIM72   00 GIL75 00000 BRB                67331   00 02113 00000
GIM73   TP GIL87 00002                    67332   11 02127 00002
GIM74   MJ 00000 GIL17                    67333   45 00000 02021
GIM75   05 00000 00000 -01 F              67334   20 04000 00000
GIM76   01 00000 00000  00 F              67335   20 14000 00000
GIM77   05 00000 00000 -01 F              67336   20 04000 00000
GIM78   02 92893 21881 -01 F              67337   17 74537 30314
GIM79   02 92893 21881 -01 F              67340   17 74537 30314
GIM80   02 92893 21881 -01 F              67341   17 74537 30314
GIM81   01 70710 67812  00 F              67342   20 16650 11714
GIM82   01 70710 67812  00 F              67343   20 16650 11714
GIM83   01 70710 67812  00 F              67344   20 16650 11714
GIM84   01 66666 66667 -01 F              67345   17 65252 52525
GIM85   03 33333 33333 -01 F              67346   17 75252 52525
GIM86   05 00000 00000 -01 F              67347   20 04000 00000
GIM87   00 00000 00000 B                  67350   00 00000 00000
GIM88   00 00000 00000 B                  67351   00 00000 00000
START                                             45 00000 00000
```

PX 71900-10-(143)

THE RAMO-WOOLDRIDGE CORPORATION
Los Angeles 45, California

FLOATING POINT SINE-COSINE

Specifications

Identification Tag:	SIN-4
Type:	Subroutine
Assembly Routine Spec:	SUB 51856 06510
Storage:	65 words total program storage
	5 words temporary storage pool used, addresses 27 b through 33 b.
	The constant pool is used by this routine.
Entrance and Exit:	RJ SUB01 SUB02 for the sine
	RJ SUB01 SUB03 for the cosine
Machine Time:	3.9 ms average, 4.8 ms maximum

Coded by:	M. Perry	May, 1956
Approved by:	W. Bauer	May 15, 1956

Description

When supplied with an argument X in SNAP form, this routine will evaluate sin X or cos X (depending on which of the two entrances is used) using a Rand Polynomial Approximation, producing the answer in SNAP form.

Programming Instructions

This routine can be inserted into a program by CMP-0 by the use of a "SUB" card in the input deck.

1. Place the double length extension of X in the accumulator.

   X must be in radians and must be in SNAP form.

2. Return jump to the subroutine. Assuming that the subroutine was assigned to region SUB00 for assembly, use either the instruction RJ SUB01 SUB02 for the sine, or the instruction RJ SUB01 SUB03 for the cosine.

3. At the time of exit from the subroutine, the double length extension of sin X (or cos X) in SNAP form will be in the accumulator.

Error Analysis

Sin X or cos X is computed to 26 bits of accuracy or to as many correct bits as there are in the Fractional portion of X, whichever is less. For $X \geq 2^{27}$, this routine substitutes zero for the argument. The alarm exit is not used.

Mathematical Method

1. Let $y = (2/\pi)X$, then $\sin X = \sin(\pi/2)(y)$

$$\cos X = \sin(\pi/2)(y + 1)$$

2. Divide y (or y + 1) into an integral part R, and a fractional part S.

3. R defines the quadrant into which X falls. Let R' be the two low order positions of R, since in binary notation, any other positions merely define a number of complete revolutions.

4. R' is a number one less than the number of the quadrant into which X falls.

5. S defines the displacement (in a position direction) within the quadrant indicated by R'.

6. Therefore, if

R' = 00	Let Z = S	first quadrant
R' = 01	Let Z = (1-S)	second quadrant
R' = 10	Let Z = (-S)	third quadrant
R' = 11	Let Z = (1-S)	fourth quadrant

PX 71900-10-(144)

7. Sin (or cos) X = sin(π/2)Z.

8. (1/z)sin(π/2)z is approximated by the Rand Polynomial Approximation Number 16, using argument z.

9. If $x < 1/2$, $(2/\pi)x$, which is in floating form, is substituted for z before doing step 10.

10. Multiply the approximation from step 8 by z giving the result, sin x (or cos x).

Range of Variable

No alarm condition is recognized by this routine. However, as X approaches $\pm 2^{27}$ the number of significant digits in Sine X (or Cosine X) approaches zero and X merely defines a number of revolutions and does not significantly designate an angle.

D		02S00	00023		00027	00	00000	00000
D		00S00	01024		02000	00	00000	00000
D		01S00	01079		02067	00	00000	00000
D		D0S00	51856		67120	00	00000	00000
D		D1S00	51911		67207	00	00000	00000
D0S00	RJ	00000	00000	ALARM	67120	37	00000	00000
D0S01	MJ	00000	00000	NORMAL EXIT	67121	45	00000	00000
D0S02	RP	20002	00S04	SIN ENTRY	67122	75	20002	02004
D0S03	TP	00013	02S04	COS ENTRY	67123	11	00015	00033
D0S04	TU	00S02	00S51	SET FOR POS	67124	15	02002	02063
D0S05	LA	A0000	00008		67125	54	20000	00010
D0S06	TM	B0000	02S00	EXP PLUS 200	67126	12	30000	00027
D0S07	LA	A0000	00001		67127	54	20000	00001
D0S08	LQ	A0000	00035		67130	55	20000	00043
D0S09	MP	Q0000	01S05	69	67131	71	10000	02074
D0S10	TP	B0000	Q0000	34	67132	11	30000	10000
D0S11	TP	Q0000	02S01		67133	11	10000	00030
D0S12	RS	02S00	01S08	EXP	67134	23	00027	02077
D0S13	SJ	00S14	00S21		67135	46	02016	02025
D0S14	SA	01S07	00000		67136	32	02076	00000
D0S15	SJ	00S18	00S16		67137	46	02022	02020
D0S16	AT	00S53	00S17		67140	35	02065	02021
D0S17	LA	Q0000	00007		67141	54	10000	00007
D0S18	TP	B0000	Q0000		67142	11	30000	10000
D0S19	TP	02S04	A0000		67143	11	00033	20000
D0S20	ZJ	00S36	00S26		67144	47	02044	02032
D0S21	TJ	01S07	00S23		67145	42	02076	02027
D0S22	CC	Q0000	A0000		67146	27	10000	20000
D0S23	TV	A0000	00S24		67147	16	20000	02030
D0S24	LA	Q0000	00000		67150	54	10000	00000
D0S25	IJ	02S04	00S28	SIN	67151	41	00033	02034
D0S26	TN	Q0000	A0000	COS	67152	13	10000	20000
D0S27	AT	01S06	Q0000		67153	35	02075	10000
D0S28	QT	01S06	02S01		67154	51	02075	00030
D0S29	CC	02S00	A0000		67155	27	00027	20000
D0S30	QJ	00S31	00S32		67156	44	02037	02040
D0S31	RS	00S51	00015		67157	23	02063	00017
D0S32	QJ	00S33	00S35		67160	44	02041	02043
D0S33	TP	01S06	A0000		67161	11	02075	20000
D0S34	ST	02S01	02S01		67162	36	00030	00030
D0S35	TP	02S01	Q0000		67163	11	00030	10000

Revised 9-14-56

```
D0S36   MP Q0000 Q0000                               67164  71 10000 10000
D0S37   SA 01S06 00001                               67165  32 02075 00001
D0S38   TP B0000 02S02               SQUARED    34   67166  11 30000 00031
D0S39   PM 01S01 01S00                               67167  24 02070 02067
D0S40   RP 20003 00S42                               67170  75 20003 02052
D0S41   PM 01S02 02S02                          69   67171  24 02071 00031
D0S42   MP B0000 02S01                          68   67172  71 30000 00030
D0S43   TP B0000 A0000               FINAL MANT33    67173  11 30000 20000
D0S44   ZJ 00S45 00S01                               67174  47 02055 02001
D0S45   SF A0000 00S54                               67175  74 20000 02066
D0S46   LA A0000 00027                               67176  54 20000 00033
D0S47   TP B0000 Q0000                               67177  11 30000 10000
D0S48   RA 02S00 00S54                               67200  21 00027 02066
D0S49   SA 01S09 00027                               67201  32 02100 00033
D0S50   CC A0000 Q0000                               67202  27 20000 10000
D0S51   RP 00000 00S01                               67203  75 00000 02001
D0S52   TN A0000 A0000                               67204  13 20000 20000
D0S53   LA Q0000 00007                               67205  54 10000 00007
D0S54   00 00000 00000                               67206  00 00000 00000
D1S00   01 51484 19000 -04   38 C9                   67207  00 02366 57351
D1S01   -4 67376 55700 -03   36 C7                   67210  77 54666 31633
D1S02   07 96896 79280 -02   35 C5                   67211  02 43150 53663
D1S03   -6 45963 71106 -01   34 C3                   67212  65 52420 76451
D1S04   01 57079 63185       33 C1                   67213  14 44176 65102
D1S05   06 36619 77225 -01   34 2 OVER PI            67214  12 13714 06667
D1S06   17 77777 77777 B        MASK                 67215  17 77777 77777
D1S07   00 00000 00034 B        28                   67216  00 00000 00034
D1S08   00 00000 00200 B        128                  67217  00 00000 00200
D1S09   00 00000 00072 B                             67220  00 00000 00072
START                                                       45 00000 00000
```

REMINGTON RAND UNIVAC

FLEXIE (RR-86 Rev. 8/56)

Type: Service Routine

I. Description:

This routine is a revised version of the original FLEXIE (RR-86). It is designed to load, by means of a Ferranti Reader, a Flexcode tape prepared on a Flexowriter in the conventional fashion for translating to bioctal. FLEXIE reads at the full speed of the Ferranti, hesitating only on insert addresses, check addresses, or when the temporary storage is full of data. FLEXIE provides a check address test and a check sum test on the data read in.

II. Tape Preparation:

The tape should be prepared in the manner described in the original FLEXIE write up. A few remarks are appropriate here however. A check address should be given following the last information on the tape and before the 7th level punch stop code. If there is no check address prior to the 7th level punch, the 7th level punch will in general be ignored.

For a check sum test of data on the tape, the following four words should be on the tape after the data to which the sum applies:

(1) insert address 75202,
(2) high order 36 bits of check sum,
(3) low order 36 bits of check sum,
(4) check address 75204.

The check sum must be the sum of all the data on the tape following the preceding check sum. The check sum will NOT be loaded into 75202 and 75203. These two words will not be disturbed at all. Note that a check sum test is performed whenever a check address of 75204 is encountered. Thus 75204 should not be used for any other check address.

III. Operating Instructions:

Since the coding is in Reco II form, all operating data will be given relative to the regions used in the code.

To load a tape, place it in the reader, turn the reader on and START at bb0. At the completion of the loading (if a 7th level punch is present on the end of the tape) a stop-re-enter bb0 is given. If there is no 7th level punch present, let the tape pass through the reader, then FORCE STOP, MASTER CLEAR, START at ee0 (ee0 will be 00022 in absolute form). FLEXIE can also be used as a subroutine to load data. To do this, place a 7th level punch after each section of the tape and perform the instruction 37 bb11 bb0 whenever it is desired to load a section of the tape.

PX 71900-10-(86)

Three different alarm conditions can occur when using FLEXIE. If one of these does occur, a tag letter is typed on the typewriter and the computer stops on a manual stop. These three conditions are now listed according to their tags.

(1) "t": FLEXIE has not been transferred to HSS correctly. START causes another transfer. If "t" occurs again, reload FLEXIE onto MD.

(2) "c": a check address has failed to check. START causes FLEXIE to ignore error and proceed to read the rest of the tape.

(3) "m": the check sum given on the tape does not agree with the sum as computed by FLEXIE. START causes FLEXIE to ignore the error.

IV. Coding:

FLEXIE requires 166 (octal) words of storage on MD, i.e., relative addresses bb0 through bb165. The region bb can be assigned any drum address except 75612 through 77777. The reason for this exception is that 76000-77777 is used for an image region of HSS and if bb has a value greater than 75612, part or all of FLEXIE will be in the image region and consequently be destroyed when FLEXIE is used. The rest of the regions used in the coding must be assigned the following values:

cc	00001
dd	00010
ee	00022
gg	00045
hh	00052
ii	00061
jj	00074
kp	00116
ws	00147
ts	00154
xx	01624

Thus one can place FLEXIE on any part of the drum by assigning a value to only one of the relative addresses, i.e., bb.

```
re bb 74520
re cc1
re dd10
re ee22
re gg45
re ii61
re hh52
re jj74
re kp116
re ws147
re ts154
re xx1624
```

```
bb0       tp  00000  76000        Entrance - store 00000
  1       tp  bb52   00000        Set up 00000
  2       rp  31777  bb4          Store HSS
  3       tp  00001  76001             on MD
  4       rp  30151  cc           Load routine
  5       tp  bb15   cc                into HSS
  6       rp  31777  bb10         Restore
  7       tp  76001  00001             HSS from
 10       tp  76000  00000             MD image
 11       rj  bb11   bb12         Subroutine exit
 12       ms  0      bb           Service routine exit - re-enter
 13       00  0      0            Flexie check sum, Lo order
 14       00  0      0                                 Hi order

 15  cc0  sp  00000  0            Compute check
 16    1  rp  20151  cc3               sum of
 17    2  sa  cc     0                 routine
 20    3  ss  bb13   44           Test check
 21    4  ej  bb14   dd                sum
 22    5  pr  0      jj2          Error, print t
 23    6  ms  0      bb4          Stop - reload routine

 24  dd0  rs  ws3    ws3          Clear check
 25    1  rs  ws4    ws4               sum cells
 26    2  tp  kp26   ii           Set up
 27    3  ef  0      kp3          Start reader
 30    4  rj  dd4    ws           One shot switch
 31    5  er  0      a            Read to A

 32    6  rp  20010  dd10         Digit
 33    7  ej  kp6    gg                test
 34   10  ej  kp5    hh           Period test
 35   11  tj  kp16   dd5          Ignore or 7th level

 36  ee0  ef  0      kp4          Stop reader
 37    1  tp  ii     a            Set up number of
 40    2  ss  kp26   17                words to transfer
 41    3  at  kp27   ee7          Set up transfer
 42    4  tp  ws1    a            Is data destined
 43    5  tj  kp     ee15              for HSS or MD?
 44    6  tv  a      ee10         Set up transfer
 45    7  rp  30000  ee11         Transfer
 46   10  tp  ts     0                 data
 47   11  tp  ws2    a            Test period code
 50   12  ej  kp22   ee17              is this ED?
```

PX 71900-10-(86)

```
 51  13     ej  kp23  ee21          is this IA?
 52  14     mj  0     bb6       Return to MD
 53  15     at  kp1   a         Modify IA to MD image
 54  16     mj  0     ee6       Jump to set up and transfer
 55  17     ra  ws1   kp2       Modify IA
 56  20     mj  0     dd2       Return to read
 57  21     tp  ws    ws1       Store IA
 60  22     mj  0     dd2       Return to read

 61  gg0    sp  ws2   1         Assemble period code
 62  1      tp  a     ws2       Store period code
 63  2      sp  ws    3         Assemble
 64  3      qa  kp17  ws            data digit
 65  4      mj  0     dd5       Return to read

 66  hh0    sp  ws2   1         Assemble period code
 67  1      at  kp21  ws2           and store
 70  2      tp  ws2   a         Period code to A
 71  3      ej  kp22  ii        Is this ED code?
 72  4      ej  kp23  ee        Is this IA code?
 73  5      ej  kp24  jj        Is this CA code?
 74  6      mj  0     dd5       Return to read

 75  ii0    tp  ws    ts        Store data word temporarily
 76  1      ra  ii    kp20      Set up for next word
 77  2      sp  ws3   44        Add assembled
100  3      sa  ws4   0             word to computed
101  4      sa  ws    0             check sum
102  5      tp  a     ws4       Store computed
103  6      la  a     44            check
104  7      tp  a     ws3           sum
105  10     tp  ii    a         Is HSS filled
106  11     ej  kp25  ee            with data?
107  12     mj  0     dd5       Return to read

110  jj0    ef  0     kp4       Stop reader
111  1      tp  ii    a7        Compute number
112  2      st  kp26  a1            of words read in
113  3      at  ws1   a16       Add to IA
114  4      ej  ws    jj7       Does this equal CA?
115  5      pr  0     jj3       Error - print c
116  6      ms  0     jj7       Stop - ignore error
117  7      tp  ws    a         Is sum check test
120  10     ej  kp30  jj12          specified?
121  11     mj  0     dd3       No, return to read
122  12     sp  ws3   44        Yes, computed check
123  13     sa  ws4   0             sum to A
124  14     ss  ts    0         Correct computed
```

125		15	ss tsl	0	check sum
126		16	ss tsl	44	Subtract lo - order part of
127		17	ej ts	dd	given sum and test difference
130		20	pr 0	jjl	Error - print M
131		21	ms 0	dd	Stop - ignore error
132		kp0	00 0	02000	Constant for test
133		1	00 0	76000	Constant
134		2	00 0	xx	Maximum block size
135		3	00 6	00000	Reader start code
136		4	00 5	00000	Reader stop code
137		5	00 0	42	Flex codes - period
140		6	00 0	72	7
141		7	00 0	66	6
142		10	00 0	62	5
143		11	00 0	64	4
144		12	00 0	70	3
145		13	00 0	74	2
146		14	00 0	52	1
147		15	00 0	37	0
150		16	00 0	100	Constant for test
151		17	00 0	7	Mask
152		20	00 0	1	Constant
153		21	00 100	00000	Constant to set up period code
154		22	20 00100	00000	ED code
155		23	40 20100	00000	IA code
156		24	40 04100	00000	CA code
157		25	tp ws	02000	Constant for test
160		26	tp ws	ts	Constants for
161		27	rp 30000	eell	set up
162		30	00 00000	75204	Sum check CA
163	ws	0	er 0	q	Clear IOA
164		1	mj 0	dd5	Return to regular read
165		2	00 0	0	Period code space

Working Space Uses:

ws 0 Assembly space for data
1 Storage for IA
2 Period code space
3 Hi-order part of computed check sum
4 Lo-order part of computed check sum

Notes: (1) Three of the working spaces are stored on MD and transferred to HSS with the rest of the program. Two of these are used initially for a one-shot clear IOA. Then they are used as indicated above throughout the rest of the program.

(2) The FLEXIE check sum cells, bbl3 and bbl4, are blank since the check sum depends on the value assigned to bb. The check

PX 71900-10-(86)

sum may be obtained from the tape assembled by Reco II as follows: load the tape and START at bb0; the computer will stop with a "t" alarm as mentioned above; at this point A-left contains the Lo-order part of the check sum and A-right the Hi-order part.