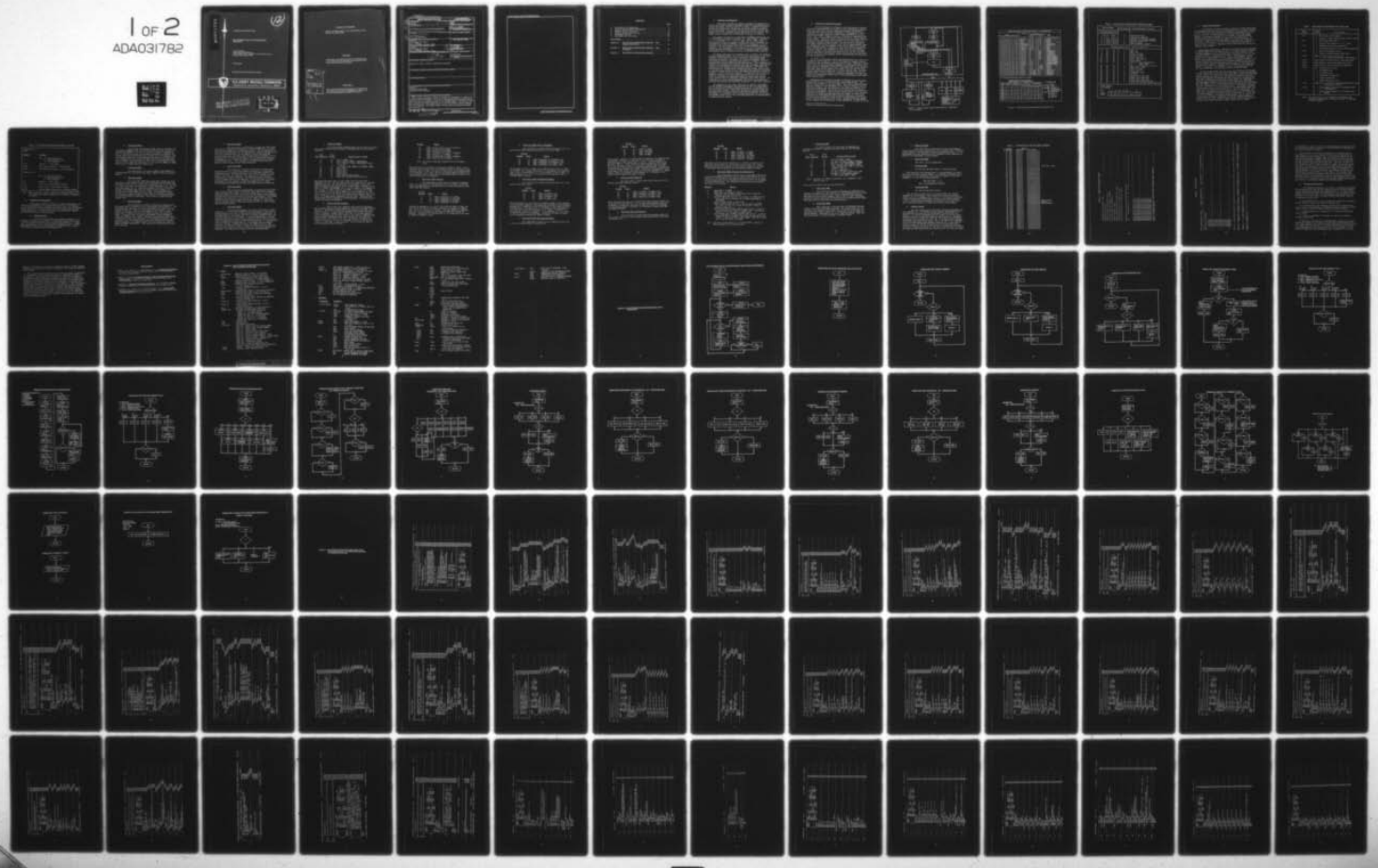
AD-A031 782 ARMY MISSILE RESEARCH DEVELOPMENT AND ENGINEERING LAB--ETC F/G 9/2
MULTIPURPOSE DIGITAL MICROPROCESSOR EMULATOR.(U)
MAY 76 J R BROOKSHIRE

UNCLASSIFIED RG-76-62 NL

1 OF 2
ADA031782



(12)

B.S.

TECHNICAL REPORT RG-76-62

MULTIPURPOSE DIGITAL MICROPROCESSOR EMULATOR

Jerry R. Brookshire
Guidance and Control Directorate
US Army Missile Research, Development, and Engineering Laboratory
Redstone Arsenal, Alabama 35809

11 May 1976

Approved for public release; distribution unlimited.

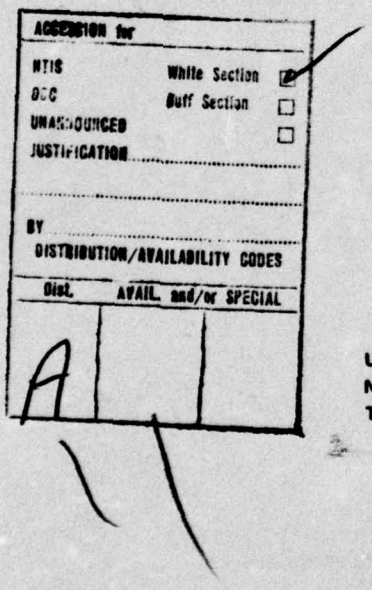
U.S. ARMY MISSILE COMMAND
Redstone Arsenal, Alabama 35809

DISPOSITION INSTRUCTIONS

DESTROY THIS REPORT WHEN IT IS NO LONGER NEEDED. DO NOT RETURN IT TO THE ORIGINATOR.

DISCLAIMER

THE FINDINGS IN THIS REPORT ARE NOT TO BE CONSTRUED AS AN OFFICIAL DEPARTMENT OF THE ARMY POSITION UNLESS SO DESIGNATED BY OTHER AUTHORIZED DOCUMENTS.



TRADE NAMES

USE OF TRADE NAMES OR MANUFACTURERS IN THIS REPORT DOES NOT CONSTITUTE AN OFFICIAL INDORSEMENT OR APPROVAL OF THE USE OF SUCH COMMERCIAL HARDWARE OR SOFTWARE.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: RG-76-62	2. GOVT ACCESSION NO.
3. RECIPIENT'S CATALOG NUMBER	
4. TITLE (and Subtitle): MULTIPURPOSE DIGITAL MICROPROCESSOR EMULATOR,	5. TYPE OF REPORT & PERIOD COVERED: Technical Report,
	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s): Jerry R. Brookshire	8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS: Commander, US Army Missile Command, Attn: DRSMI-RG, Redstone Arsenal, Alabama 35809	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS: Commander, US Army Missile Command, Attn: DRSMI-RPR, Redstone Arsenal, Alabama 35809	12. REPORT DATE: 11 May 1976
	13. NUMBER OF PAGES: 117
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office): 119 p.	15. SECURITY CLASS. (of this report): Unclassified
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Emulator
Digital microprocessor
Guidance control computer

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This report describes an emulator designed and implemented to support the initial software development effort for a multipurpose digital microprocessor, and which also is to provide a vehicle for experimenting with hardware and software architectural changes proposed for the microprocessor. The initial testbed application for the microprocessor is to be the guidance control computer in the T-6 missile. The flight control software will be developed utilizing this emulator.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

400469

CONTENTS

1. Introduction and Background

This report describes an emulator designed and implemented to support the initial software development effort for a multipurpose digital microprocessor, and which also is to provide a vehicle for experimenting with hardware and software architectural changes proposed for the microprocessor. The initial test-bed application for the microprocessor is to be the guidance control computer in the T-6 missile. The flight control software will be developed utilizing this emulator.

The emulator is written in a variation of FORTRAN allowing the use of "structured" statements, including: IF...THEN...ELSE...; DO UNTIL (...)...; DO WHILE (...)...; and IF...THEN...OR IF...THEN...ELSE... This program is first run through a preprocessor which translates the statements into standard FORTRAN, providing the capability to run the emulator on any host computer which has a standard FORTRAN compiler available. The utilization of the emulator for the T-6 missile flight control software development is to be on the Missile Computer Software and Hardware Center's Raytheon R-520 and the MICOM CDC 6600 computers.

The design objectives of the emulator were to support the immediate software development effort for the T-6 missile application of the microprocessor, while retaining as much flexibility for proposed-change evaluation as possible. To support the software development effort, it was essential that the emulator perform exactly as though it were the microprocessor hardware executing microinstructions. Both considerations, fidelity of microinstruction execution and ease of change, supported the position that the emulator should be highly modular. Additional considerations which influenced the emulator design include: ease of set-up and operation; varying levels of detailed output available optionally to support different types of runs; and maximum portability to allow the emulator to run on other host computers. A single run control card was designed to define the desired characteristics of each run, and provides the mechanisms for specifying simulated memory preset values, output options, and run termination conditions (other than errors). For portability, the host-computer-dependent portions of the program were minimized and isolated for simplified conversion to other host computers.

The remainder of this report describes the specific requirements and the specific programming techniques used to meet the requirements, plus a description of some potential extensions to or generalizations from the current design. Since early applications software design and experimentation are often crucial during computer hardware definition and design, it is believed that the concepts provided here can provide an extremely useful tool in the continuing application of microprocessors to weapons control systems.

2. Microprocessor Hardware Organization

From the viewpoint of software, the microprocessor* consists of a 16-bit (reconfigurable up or down in 4-bit increments) arithmetic logic unit (ALU) with a 16-word random access memory (scratchpad), a 32-operation instruction set, expandable to 58 operations by interpretation of a carry-control bit, and further expandable by a 3-bit instruction modifier (Figures 1 and 2). There are two associated memories: 1024 - 16-bit words of main memory; and 1024 - 48-bit words of control memory. Normal execution of a macroinstruction begins with the decoding of the main memory word pointed to by the first word in the scratchpad (software program counter - PC) into two 8-bit addresses. The leftmost eight bits are placed in the control memory address register (H-register), and the rightmost eight bits are placed in the main memory address register (C-register). Both the H- and C-registers are 10 bits in length: The C-register is wired such that this decoding operation causes the two high-order bits to be on, so that the references to main memory following the decode will be offset by 1400 octal. The top two bits of the H-register can be controlled from the execution of a microinstruction.

The 48-bit control memory words (microinstructions) are broken into 25 distinct fields (Table 1). Six of the fields are encoded, and the remaining 19 are considered as discrete control bits. Eighteen of these control bits are now functionally assigned, but additional control functions could be implemented if required by grouping some of the functions under additional encoded fields; i.e., where three bits are now three functions, three bits encoded could be used to control eight functions. Current encoded fields consist of: the 5-bit operation code, a 3-bit modifier code, two 4-bit operands (scratchpad addresses), a 10-bit next control word (H-register) field, and a 3-bit analog/digital (A/D) multiplexer address field.

Macroinstruction execution proceeds by decoding the control memory word pointed to by the H-register after a main-memory decode. The design of the hardware is such that some of the control functions specified by discrete bits are performed first. Then the operation code is executed, followed by the disposition of the result as directed by the instruction modifer, and the functions of the remaining discrete control bits are performed. One of the control bits is used to enable the loading of the H-register from the 10-bit next-H field. Thus, the microinstructions comprising a macrooperation are linked. The last microinstruction of a macrosequence will contain the address of a microinstruction which initiates the next decode of a main memory word through the PC, and then increments the PC by one. Therefore, the main memory control program will be sequential, unless modified by a reload of the PC.

---

*Monolithic memories 5701/6701 microprocessor.

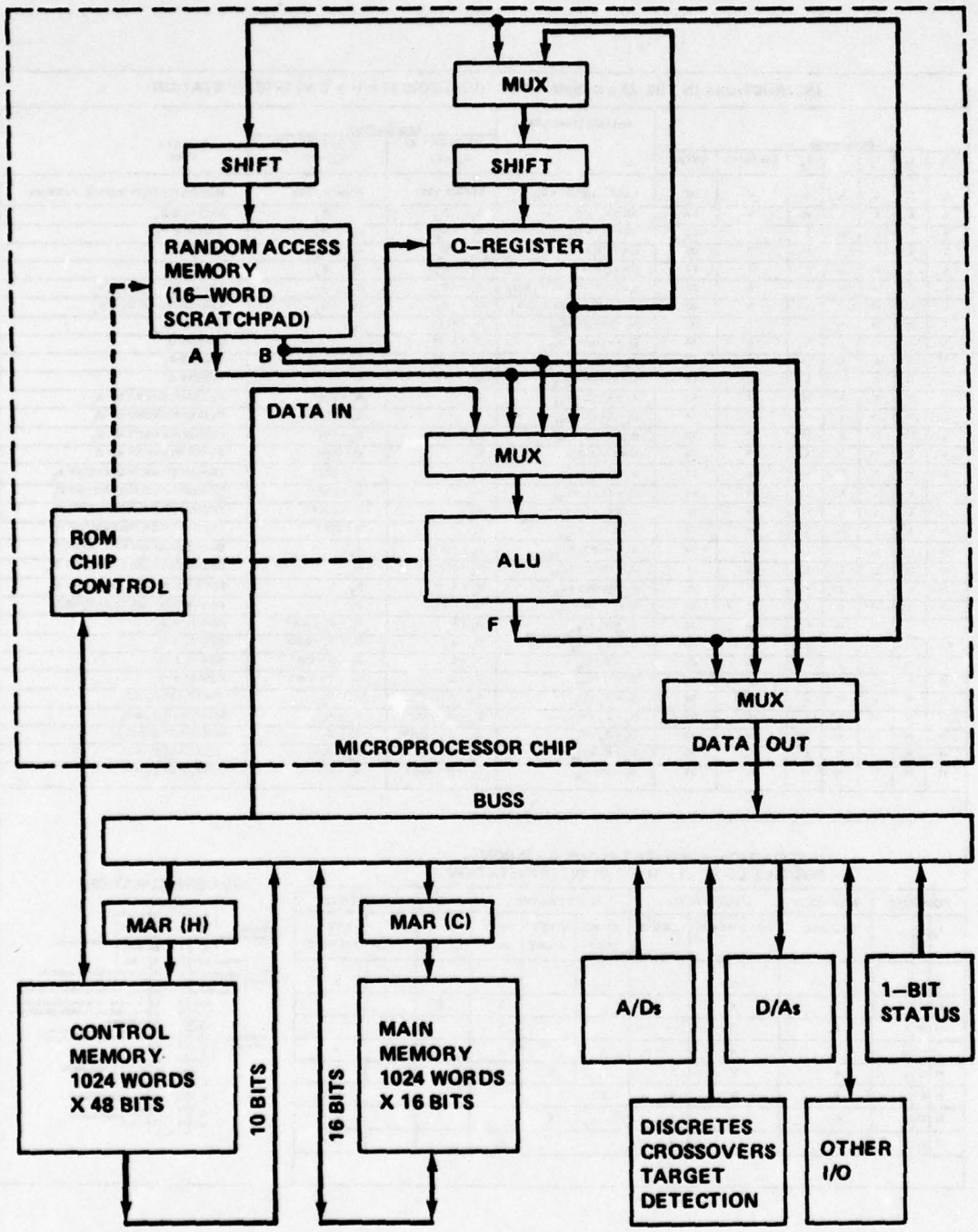


Figure 1. Multi-purpose digital microprocessor simplified block diagram.

INSTRUCTIONS IN THE 32 x 9 ROM – POSITIVE LOGIC (1 = H ≈ 3 V) INTERPRETATION

ROM WORD I_7	I_6	I_5	I_4	I_3	DECIMAL	OCTAL	ALU INSTRUCTION	ALU OUTPUT: NO CARRY IN (C_N = L)	ALU OUTPUT: WITH CARRY IN (C_N = H)	TYPICAL USES
L	L	L	L	L	0	00	LLLL + HHHH + C_N	FORCE 1111	FORCE 0000	INITIALIZATION (FORCE 1's OR 0's)
L	L	L	L	H	1	01	AND A_I & B_J	A_I B_J	A_I B_J	AND A_I & B_J
L	L	L	H	L	2	02	AND D_I & B_J	D_I B_J	D_I B_J	AND D_I & B_J
L	L	L	H	H	3	03	OR A_I & B_J	A_I B_J	A_I B_J	OR A_I & B_J
L	L	H	L	L	4	04	OR D_I & B_J	D_I B_J	D_I B_J	OR D_I & B_J
L	L	H	L	H	5	05	EXCLUSIVE OR A_I & B_J	A_I B_J	A_I B_J	EXCLUSIVE OR A_I & B_J
L	L	H	H	L	6	06	EXCLUSIVE OR D_I & B_J	D_I B_J	D_I B_J	EXCLUSIVE OR D_I & B_J
L	L	H	H	H	7	07	A_I + HHHH + C_N	A_I + 1111	A_I	INVERT A_I
L	H	L	L	L	8	10	D_I + HHHH + C_N	D_I + 1111	D_I	INVERT D_I
L	H	L	L	H	9	11	B_J + HHHH + C_N	B_J + 1111	B_J	INVERT B_J
L	H	L	H	L	10	12	Q + HHHH + C_N	Q + 1111	Q	INVERT Q
L	H	L	H	H	11	13	A_I + LLLL + C_N	A_I	A_I + 0001	2's COMPLEMENT OF A_I
L	H	H	L	L	12	14	D_I + LLLL + C_N	D_I	D_I + 0001	2's COMPLEMENT OF D_I
L	H	H	L	H	13	15	B_J + LLLL + C_N	B_J	B_J + 0001	2's COMPLEMENT OF B_J
L	H	H	H	L	14	16	Q + LLLL + C_N	Q	Q + 0001	2's COMPLEMENT OF Q
L	H	H	H	H	15	17	A_I + LLLL + C_N	A_I	A_I + 0001	TRANSFER OR INCREMENT A_I
H	L	L	L	L	16	20	D_I + LLLL + C_N	D_I	D_I + 0001	TRANSFER OR INCREMENT D_I
H	L	L	L	H	17	21	B_J + LLLL + C_N	B_J	B_J + 0001	TRANSFER OR INCREMENT B_J
H	L	L	H	L	18	22	Q + LLLL + C_N	Q	Q + 0001	TRANSFER OR INCREMENT Q
H	L	L	H	H	19	23	A_I + HHHH + C_N	A_I + 1111	A_I	DECREMENT OR TRANSFER A_I
H	H	L	L	L	20	24	D_I + HHHH + C_N	D_I + 1111	D_I	DECREMENT OR TRANSFER D_K
H	L	H	L	H	21	25	B_J + HHHH + C_N	B_J + 1111	B_J	DECREMENT OR TRANSFER B_J
H	L	H	H	L	22	26	Q + HHHH + C_N	Q + 1111	Q	DECREMENT OR TRANSFER Q
H	L	H	H	H	23	27	A_I + B_J + C_N	A_I + B_J	A_I + B_J + 0001	ADD A_I & B_J
H	H	L	L	L	24	30	D_I + B_J + C_N	D_I + B_J	D_I + B_J + 0001	ADD D_I & B_J
H	H	L	L	H	25	31	A_I + Q + C_N	A_I + Q	A_I + Q + 0001	ADD A_I & Q
H	H	L	H	L	26	32	D_I + Q + C_N	D_I + Q	D_I + Q + 0001	ADD D_I & Q
H	H	L	H	H	27	33	A_I + B_J + C_N	A_I – B_J – 0001	A_I – B_J	SUBTRACT A_I & B_J
H	H	H	L	L	28	34	B_J + A_I + C_N	B_J – A_I – 0001	B_J – A_I	SUBTRACT B_J & A_I
H	H	H	L	H	29	35	D_I + B_J + C_N	D_I – B_J – 0001	D_I – B_J	SUBTRACT D_I & B_J
H	H	H	H	L	30	36	B_J + D_I + C_N	B_J – D_I – 0001	B_J – D_I	SUBTRACT B_I & D_I
H	H	H	H	H	31	37	D_I + Q + C_N	D_I – Q – 0001	D_I – Q	SUBTRACT D_I & Q

A_I: FIRST OPERAND; B_J: SECOND OPERAND; D_I: DATA IN; Q: 16-BIT Q REGISTER

INSTRUCTION MODIFIERS IN THE 8 x 8 ROM – POSITIVE LOGIC (1 = H ≈ 3 V) INTERPRETATION

ROM WORD I_2	I_1	I_0	ROM WORD DECIMAL	LOAD CONTROL: LOAD RAM B_J	LOAD Q	SHIFT CONTROL: SHIFT LEFT	SHIFT RIGHT	DON'T SHIFT	DATA OUT CONTROL: A LATCH	B LATCH	ALU OUTPUT F
L	L	L	0	X				X			X
L	L	H	1	X				X	X		
L	H	L	2	X				X		X	
L	H	H	3	X		X					X
H	L	L	4	X			X				X
H	L	H	5	X	X	X					X
H	H	L	6	X	X		X				X
H	H	H	7		X			X			X

PIN CONFIGURATION

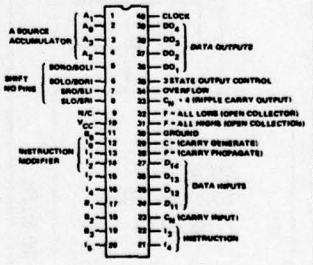
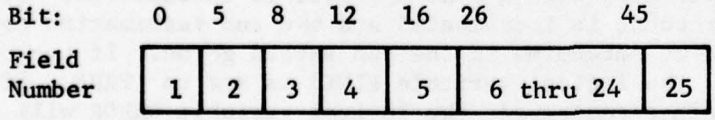


Figure 2. Multipurpose microprocessor instruction set.

TABLE 1. MULTIPURPOSE MICROPROCESSOR INSTRUCTION FORMAT

Bit Positions			Field Length	Field Name/Control Function
Field	Decimal	Octal		
1	0-4	0-4	5	Instruction Code
2	5-7	5-7	3	Instruction modifier
3	8-11	10-13	4	Operand A (scratchpad address)
4	12-15	14-17	4	Opernad B (scratchpad address)
5	16-25	20-31	10	Next H address
25	45-57	55-57	3	A/D multiplexor address
				Discrete Control Bits: (Action if Bit = 1)
6	26	32	1	SR1 = 0
7	27	33	1	SR2 = SR1
8	28	34	1	H Reg = CM addr
9	29	35	1	Main (C) = Buss
10	30	36	1	Buss = Main (C)
11	31	37	1	Buss = ALU data out
12	32	40	1	Fetch enable
13	33	41	1	Data in = Input discretes or micad (KAD)
14	34	42	1	SR1 = 1 if 4 data out = All 0's
15	35	43	1	LSB of buss = SR1
16	36	44	1	LSB of H = SR2
17	37	45	1	Carry (CN) = 1
18	38	46	1	Telemetry = Buss
19	39	47	1	C = Buss
20	40	50	1	Select D/A 1 and clock
21	41	51	1	Select D/A 2 and clock
22	42	52	1	Select D/A 3 and clock
23	43	53	1	Clock A/D mux addr register
24	44	54	1	Clock A/D conversion start (per bit)

Control Memory
Word Format:



3. Emulator Control Structure

The emulator, as now structured, consists of a main program and 25 subprograms which are used for initialization and run control; instruction and control-function simulation, diagnostic and statistical output, and the provision of machine-dependent (Raytheon 520) direct code support. It was also determined to be necessary to provide FORTRAN masking and shifting operations for bit access and manipulation. An exclusive OR function has also been added to the function library of the R-520. These direct code and FORTRAN extensions tend to reduce the portability of the system, but they were minimized and simplified so that the effort required to duplicate these capabilities on another host computer can be minimal.

All microprocessor memory, registers, and data paths are represented by FORTRAN COMMON variable names in order to minimize parameter passing in calling sequences. The organization of the R-520 computer is such that REAL FORTRAN variables are represented internally by 48 bits, and integers have 24-bit representation. Thus, a microprocessor control memory word is represented by a real variable name (CNTMEN), and all other microprocessor entities are represented by integer variables. The implicit variable-type identifiers of standard FORTRAN are used exclusively, while at the same time maintaining at least an abbreviation of the name of the elements, preceeded where necessary by an integer-type letter (I through N). For example, the two 1-bit registers, SR1 and SR2, are identified in the emulation program as MSR1 and MSR2.

The main program reads the run control card (Table 2) and proceeds to step through all the run-initiation subprograms. These are described in detail later, but consist of initializing all simulated microprocessor memory, registers and data paths, and reading microinstructions and main memory sequence instructions and data values into the appropriate locations of simulated microprocessor memory (Table 3). Execution is then initiated by entering an outer loop (DO WHILE .NOT.FINIS(hed). AND..NOT.ERROR) in which a main memory access and decode is accomplished. An inner (macro) loop is then entered which is controlled by the contents of the H-field of the microinstructions. The emulator simulates the execution of all functions specified by the microinstructions, and falls out of the inner loop when a next-H = zero is encountered. At this point, the macrocount is incremented and the run termination conditions are examined to determine if the run should go on. If the run is to stop normally, the logical variable FINIS is set to .TRUE.; if a fatal error has been encountered, the logical variable ERROR will have been set to .TRUE. Either condition will cause the simulated run to stop, and a final statistical output routine will be called.

TABLE 2. MULTIPURPOSE MICROPROCESSOR RUN CONTROL CARD

Card Columns	Octal Value	Interpretation
1-4	1	Pre-clear control memory.
	2	Pre-set control memory, NO-OP, next H = here.
5-8	1	Pre-clear main memory
	2	Pre-set main memory, H = 0, C = here.
9-12	1	Minimum output during execution
	2	Extensive diagnostic output during execution.
13-16	0	No initialization output.
	1	Output simulated memory after load.
17-20	XXXX	Ten-bit initial value of PC. (i.e., starting address of main memory)
21-24	XXXX	Output control-interval time-octal (sec)
25-28	XXXX	Output control-microinstruction count.
29-32	XXXX	Output control-selected macroaddress.
33-36	Run	Termination criteria
	1	H-address stop
	2	C-address stop
	3	Simulated elapsed time
	4	Real elapsed time
	5	Macroinstruction count
37-40	XXXX	Octal termination value associated with above termination criteria.
41-44	1	Load memory cards are in contiguous octal representation
	2	Load control memory by 25 discrete octal fields

Note: Run control card can be expanded to accommodate more variable considerations - currently the card is read with a (20 04) format.

TABLE 3. MULTIPURPOSE MICROPROCESSOR MEMORY LOAD CARD

N-RUN (11) = 1: Card	
Columns	Contents
1	Key: 1 = main memory load 2 = control memory load 9 = load completed
9-12	Address: 4 octal digits
15-20	Main memory contents: 6 octal digits
21-36	Control memory contents: 16 octal digits
N-RUN (11) = 2:	
1	Key: 1 = Main memory load 2 = Control memory load 9 = Load completed
3-6	Address: 4 octal digits
8-10	Left 8 bits of main memory (octal)
11-13	Right 8 bits of main memory (octal)
15-33	Control memory instruction, by fields (octal)

Note: Memory load card could be extended to register loads or other size memory word loads by adding keys to columns 1 through 8, as desired.

4. Emulator Routine Descriptions

This section describes in detail the actions performed by the individual subprograms of the microprocessor emulator. The sequence of the descriptions follows generally the order in which the routines are called, except that the more general purpose and widely used routines are described last. Flowcharts, listings, and detailed data element definitions are given on the appendices.

a. Subroutine INIT

This routine initializes all nonmemory elements of the simulated hardware to zero, except for scratchpad word one (PC), which is set to the value read from the run control card [NRUN(5)]. In addition, the emulator control variables FINIS and ERROR are set to .FALSE., and operation count and time accumulation variables are set to zero. This routine is called one time per emulator run.

b. Subroutine SETCM

This routine initializes all 1024 words of simulated control memory (CNTMEM), based on an initialization key read from the run control card (NRUN(1)). If the key is 1, all of memory is cleared to zero; if the key is 2, each word in memory is preset to a NO-OP in the op code field, and the H-field is set to point to its own address. With this setting, execution of a microprogram will hang in a one-word loop if it inadvertently gets outside the area where the microprogram under test is loaded. Any other value of the key on the run control card will cause an error message and will abort the run by setting ERROR to .TRUE. This routine is called once per emulator run.

c. Subroutine SETMN

This subroutine is the same as SETCM, except NRUN(2) is the applicable key, and presets main memory (MAINM) rather than control memory.

d. Subroutine LOADM

This subroutine reads data cards which contain the octal address and octal contents of all words of simulated control memory and main memory which are to be loaded for an emulation run. A key in column 1 is used to determine the type memory a given card applies to: key = 1 loads main memory, key = 2 loads control memory, and key = 9 signifies the load is completed; any other key causes an error message to be printed, and the flag ERROR is set to .TRUE. The load cards may contain the 16- or 48-bit contiguous octal value to be placed in memory, or may contain separate fields for each field of the macroinstructions and microinstruction, as preferred. This routine is called once per emulator run.

e. Subroutine DCODM

This subroutine is called to initiate the execution of a macroinstruction by accessing a word of main memory and decoding it. The main memory word accessed is the one whose address is currently in the Program Counter (PC - word one of the simulated scratchpad). Once the address is retrieved, the PC is incremented by one. Then the main memory word is broken into two 8-bit fields utilizing the subroutine GETMM. If an error flag is returned from GETMM, an error message is output and ERROR is set to .TRUE. Otherwise, the left-most 8 bits are incremented by 1400 octal and placed in the simulated H-register, and the righ-most 8 bits are incremented by 1400 octal and placed in the simulated C-register. The original values of the two halves are stored in MINSTR and MOPND respectively. This subroutine is called once for each Macro executed.

f. Subroutine GETMM

This subroutine is generalized to extract any given field from a 16-bit simulated word located in a 24-bit R-520 word. If therefore has five parameters in its calling sequence: (1) the address of the word to be extracted from, (2) the starting bit, decimal 0 through 15, of the field to be extracted, (3) the number of bits to be extracted, (4) the address that the field, right adjusted and zero-filled, is to be stored in, and (5) an error flag which is set to zero when no error is encountered, or to an integer to identify the parameter which contained a recognizable error. This subroutine is normally callled twice per macro, or once for each desired field extraction.

g. Subroutine DCODC

This subroutine is called to break down a 48-bit microinstruction into its constituent fields (currently 25) to facilitate interpretation and simulated execution of the specified operations and control functions. The selected 48-bit word is that pointed to by the index in the simulated H-register. The extraction of the fields is accomplished by calling GETCM, and the fields are placed in a 25-word vector in COMMON identified as MICRO(25). If an error flag is returned from GETCM, an error message is output and ERROR is set to .TRUE. The subroutine is called once for each microinstruction to be executed.

h. Subroutine GETCM

This subroutine is generalized to extract any given field from a 48-bit (60-bit in CDC version) real word, and return the result right-justified, zero-filled, in a 24-bit integer word. It is therefore currently limited to a 24-bit extraction, but the largest field currently required from a microinstruction is the 10-bit next-H field. The calling parameters are identical to those listed for GETMM, except that the starting bit may be any decimal number from 0 through 47. This subroutine is normally called 25 times per microinstruction, or once for each desired field extraction.

i. Subroutine EXECM

This subroutine manages the complex execution of a microinstruction. It first calls PRBIN for the application of those control functions which occur prior to instruction interpretation/execution. It then determines if the register SR2 is to cause the operation code to be modified, and does so if indicated. It then determines which of seven functional areas of execution that the operation code falls in, and then calls the appropriate routine to execute that operation. If the operation code is outside the allowed range (0 - 31 decimal), then an emulator error has occurred, a message is output, and ERROR is set to .TRUE. Otherwise, the subroutine BINRY is called to interrupt and act upon the binary control fields which apply in time afer instruction execution is completed. This subroutine is called once per microinstruction.

j. Subroutine PRBIN

The microprocessor hardware design is such that the following control functions are performed, if enabled, prior to instruction execution:

Field (Ref. Figure 2)	Octal Bit No.	Action if Bit = 1 (On)
6	32	SR1 = 0 (MSR1 = 0)
10	36	Buss = Main(c) (MBUSS - MAINM(MEMC))
13	41	Buss = DATA IN = (Input Discretes) OR (A/D Output)
15	43	LSB or Buss = SR1 (MBUSS = OR (MBUSS, MSR1))
17	45	Carry (Cn) = 1
20	50	Select D/A 1
21	51	Select D/A 2
22	52	Select D/A 3
23	53	Clock A/D Mux Addr Register
24	54	Clock A/D conversion Start (per Bit.)

The subroutine checks each of these fields [micro(6) micro(10),... MICRO(23)] in turn: For each that is set (i.e., equal to 1), the indicated operation (or test and operation) is performed. Most of the rquired actions are the setting of one variable equal to another (i.e., MSR1 = 0), MBUSS = MAINM(MEMC), etc.), but not always. For example, if bit 43 octal = 1, MSR1 is ORed to the MBUSS, so that the only change that if MSR1 = 1 and the LSB of MBUSS is 0, it is changed to 1. The last four controls cause special keys (KDA for D/A, KAD for A/D) to be set at nonzero (1, 2, or 3) for the appropriate D/A register select, or equal to the MUX address [MICRO(25)] for A/D input. Timing for A/D setting delays must be managed by the microprogram. This subroutine is called once per microinstruction.

k. Subroutine BOLN (Boolean)

This is the first operation code execution subprogram, and is called when the effective operation code is in the range 00 to 06 inclusive. The operation specified are all Boolean except that generated by OP code 0. In this and all subsequent OP code execution routines, a temporary variable ITMP is first set to the value resulting from the operation, and the final effect on the status of registers and output lines is determined later, after analysis of the instruction modifier field. The status of the carry in register ICN affects the result of OP code 0. If ICN = 0, then ITMP = 177777, or all bits on. If ICN = 1, then ITMP = 0 (all bits off). For the other OP codes in this subroutine, ICN has no effect.

OP Code	Result	
00	ITMP = 177777 if ICN = 0; = 0 if ICN = 1	
01	ITMP = (A OPND) and (B OPND)	
02	ITMP = (Data in) and (B OPND)	
03	ITMP = (A OPND) or (B OPND)	Inclusive
04	ITMP = (Data IN) or (B OPND)	or
05	ITMP = (A OPND) XOR (B OPND)	Exclusive
06	ITMP = (Data IN) XOR (B OPND)	or

Note: The data IN lines are represented by the variable IDIN.

This subroutine then examines the instruction modifier field. If the modifier value is seven, then the Q-register (MICQ) is set equal to the resultant value of ITMP, and control is returned to the calling program. If the modifier is not seven, then the scratchpad address pointed to by the B-operand is set to ITMP, and the subprogram INMOD is called to complete the effects of the modifier.

1. Subroutine INVRT (Invert)

This subprogram is called when the effective operation code is in the range 07 to 10 inclusive (07 to 12 octal). In the emulator, the operations are performed first, and then the effects of the carry (ICN) are applied.

OP Code Decimal	Octal	Result
7	07	ITMP = Complement of (A OPND)
8	10	ITMP = Complement of (Data IN)
9	11	ITMP = Complement of (B OPND)
10	12	ITMP = Complement of (Q REG)

Then ICN is examined if = 0, 177777 (all ones) are added to ITMP, and the bits to the left of the 16-bit simulated word length are masked off. If ICN = 1, ITMP is not changed. Then the modifier is examined, and, if = 7, the Q-register is set equal to ITMP, and control is returned. Otherwise, the B-operand word in the scratchpad is set equal to ITMP, and INMOD is called.

m. Subroutine CMPLT (Two's Complement)

This subroutine is entered when the OP code is 11 to 14 and performs almost identical operations to those of INVRT:

OP Code		
Decimal	Octal	Result
11	13	ITMP = Complement of (A OPND) + ICN
12	14	ITMP = Complement of (Data IN) + ICN
13	15	ITMP = Complement of (B OPND) + ICN
14	16	ITMP = Complement of (Q REG) + ICN

In this case, if ICN = 0, ITMP is not affected; and if ICN = 1, ITMP is increased by 1, thereby performing a 2's complement on the selected data. Then, if the modifier is 7, the Q-register is set to ITMP. If the modifier is other than 7, the value of the B-operand is set to ITMP and INMOD is called.

n. Subroutine TRINC (Transfer/Increment)

This subroutine is called when the OP code is 15 to 18, with the following results:

OP Code		
Decimal	Octal	Result
15	17	ITMP = (A OPND) + ICN
16	20	ITMP = (Data IN) + ICN
17	21	ITMP = (B OPND) + ICN
18	22	ITMP = (Q REG) + ICN

Here, when carry is 0, the effect is a simple transfer of the contents of the designated operand to the designated B-operand or the Q-register, depending on the modifier. Note that operation 17 (21 octal) will have a no OP effect if the modiier is 0. The emulator still performs the load ITMP from B and stores ITMP back in B, however. If carry = 1, the operation includes an increment. If the modifier is 7, the value in ITMP is stored in the Q-register. Otherwise, the value is stored in the designated B-operand and INMOD is called.

p. Subroutine DECTR (Decrement/Transfer)

This subprogram is called when the operation code is 19 to 22, with results similar to TRINC above:

OP Code		
Decimal	Octal	Result
19	23	ITMP = (A OPND)
20	24	ITMP = (Data IN)
21	25	ITMP = (B OPND)
22	26	ITMP = (Q REG)

When the basic operation of loading ITMP is completed, the value of carry is examined. If carry = 0, than all 1's (177777 octal) is added to ITMP, and ITMP is then masked back to 16 bits. Thus the effect is to decrement ITMP by 1, since the action constitutes addition of a minus one. This technique is used in the emulator because that is the way the operation is performed in the microprosessor hardware. If carry = 1, no further change is made yet to ITMP. Here again is a potential no-OP, in OP code 21 (25 octal), when carry is on and the modifier is 0. Again, ITMP is placed in Q-register if the modifier is 7; otherwise, it is placed in the B-operand and INMOD is called.

q. Subroutine SUM (Addition)

This subroutine is entered when the OP code is 23 to 25, and the following operations result:

OP Code		
Decimal	Octal	Result
23	27	ITMP = (A OPND) + (B OPND) + ICN
24	30	ITMP = (Data IN) + (B OPND) + ICN
25	31	ITMP = (A OPND) + (Q REG) + ICN
26	32	ITMP = (Data IN) + (Q REG) + ICN

Here, the effect of carry is to increase the sum by one if on (ICN = 1), or no effect if off (ICN = 0). If the instruction modifier value is 7, Q-register is loaded with the value in ITMP, and control is returned to the calling program. Otherwise, the B-operand is loaded from ITMP and INMOD is called.

r. Subroutine DIFF (Subtraction)

This subroutine is called when the operation code is 27 to 31, and is the last of the operation analysis/execution subroutines. The effects are:

OP Code		
Decimal	Octal	Result
27	33	ITMP = (A OPND) - (B OPND)
28	34	ITMP = (B OPND) - (A OPND)
29	35	ITMP = (Data IN) - (B OPND)
30	36	ITMP = (B OPND) - (Data IN)
31	37	ITMP = (Data IN) - (Q REG)

The effect carry has on this operation is to reduce the result by one when carry is off (ICN = 0), or no effect when on (ICN = 1). Again, ITMP is stored in the Q-register if the instruction modifier is minus 7; otherwise, ITMP is stored in the B-operand and INMOD is called.

s. Subroutine INMOD (Instruction Modification)

This subprogram is called from the applicable instruction interpretation/execution routine when the instruction modifier is not 7, but 0 to 6. At this point, the value produced by execution of the operation has been placed in the scratchpad word pointed to by the B operand of the microinstruction. The following actions are now taken based on the value of the modifier:

Modifier	Action
0	(Data OUT) = (B OPND)
1	(Data OUT) = (A OPND) ('A' Latch).
2	(Data OUT) =old (B OPND), before execution of operation changed its value ('B' Latch).
3	Shift (B OPND) left one bit - zero enters (B OPND) from the right, and the bit shifted off is inverted and ORed to SR1 (MSR1). Result is also placed in (Data OUT).
4	Shift (B OPND) right one bit - a one bit enters (B OPND) from left, and the bit shifted off is lost. Result is placed in Data Out.
5	Shift (Q-REG and (B OPND) together left one bit - a one bit enters q from right - MSB of (B OPND) → SR1. Data OUT is then set to the resultant (B OPND) value.
6	Shift (Q-REG) and (B OPND) together right one bit - a one bit enters the MSB of (B OPND), and the bit shifted off of the Q-REG is inverted and ORed to SR1 (MSR1). Data OUT is then set to the new value of (B OPND).

Note: Data OUT is represented by the variable IDOUT. Control is then returned to the calling routine.

t. Subroutine BINRY

The binary control bits which were not examined for required action prior to operation code execution are now evaluated. They are:

Field (Ref. Figure 2)	Octal Bit No.	Action if BIT = 1 (on)
7	33	SR2 = SR1 (MSR2 = MSR1)
8	34	H - REG = CM (addr) (MEMH = MICRO(5))
9	35	MEM (C) = Buss (MAINM (MEMC) = MBUSS)
11	37	Buss = ALU Data OUT (MBUSS = IDOUT)
12	40	Fetch (Confirm that H REG = 0; if not, ERROR = .TRUE.)
14	42	SR1 = 1 if Data OUT = 0
16	44	H - REG = SR1 ORed to LSB of H - REG
18	46	Telemetry = Buss (ITELM = MBUSS)*
19	47	C - REG = Buss (MEMC - MBUSS)

*Note: Telemetry bit effect is inverted, i.e., telemetry = buss when bit $46_8 = 0$.

This routine is called once per microinstruction.

u. Subroutine FINAL

This subroutine is entered following the completion of a complete macrooperation, and evaluates the run-termination criteria which were setup by the run control card. If the termination criteria are satisfied, then the logical variable FINIS is set to .TRUE., otherwise no action is taken, and control is returned to the calling routine (MAIN). This subroutine is called once per macro-operation.

v. Subroutine STATS

This subroutine is entered after the simulated microprocessor run has been terminated or aborted. The routine prints out the contents of all registers, data paths, and the 16-word scratchpad, plus the simulated run time, number of microinstructions, and the number of macros executed. Additional output can be added as desired. When control is returned to the main program, the program stops.

w. Subroutine REPT1

This subroutine is called when the level of diagnostic output required is high as indicated on the run control card. Currently, outputs result in the execution of individual microinstruction. It is anticipated that this subroutine will be expanded as more experience is gained in the development of microprograms.

x. Subroutine REPT2

Stub only - not yet implemented.

y. Subroutine XOR

Not used for CDC 6000 version: CDC function XOR.

This subroutine was developed prior to the availability of exclusive OR in the R-520 function library. It performs an exclusive OR on the second or third parameters, and returns the results in the first parameter. It uses the following algorithm:

```
CALL XOR (IANS, J, K)
IANS = ((J)AND(K))OR((J)AND(K)).
```

z. Subroutine FLEX

Not used for CDC 6000 version.

This subroutine contains all the required direct code (R-520 FLEXTRAN) for the emulator. Two functions are currently performed, based on the key in parameter one: If key = 1, the subroutine converts a right-adjusted real (double-word) variable to an integer variable. If K = 2, the subroutine increments the H-field (bit 31 octal) of a 48-bit microinstruction by 1, used in presetting control memory.

5. Preliminary Results

The basic design goal of providing a software development and test capability for the digital control computer application of the microprocessor has been met, and the emulator is now serving this capacity. It has been determined that the ability to vary the intermediate output over a very wide range has been difficult to implement, but further work is planned in this area. The current version allows essentially three options of output: (a) a full printout of simulated control and main memory after loading (Table 4), (b) a very detailed printout of the intermediate and final results of each instruction execution (Table 5), and (c) a final status of all registers and data paths after termination (Table 6). The last of these three is currently not under run card control, but the other two are. It is not yet known

TABLE 4. MICROPROCESSOR SIMULATED MEMORY CONTENTS

LOCATION		CONTENTS		
DECIMAL	OCTAL	MAIN MEMORY	CONTROL MEMORY	
38	0046	00000046	4200001142000000	
39	0047	00000047	4200001162000000	
40	0050	00000050	4200001202000000	
41	0051	00000051	4200001222000000	
42	0052	00000052	4200001242000000	
43	0053	00000053	4200001262000000	
44	0054	00000054	4200001302000000	
45	0055	00000055	4200001322000000	
46	0056	00000056	4200001342000000	
47	0057	00000057	4200001362000000	
48	0060	00000060	4200001402000000	
49	0061	00000061	4200001422000000	NO-OP, NEXT H = HERE
50	0062	00000062	4200001442000000	
51	0063	00000063	4200001462000000	
52	0064	00000064	4200001502000000	
53	0065	00000065	4200001522000000	
54	0066	00000066	4200001542000000	
55	0067	00000067	4200001562000000	
56	0070	00000070	4200001602000000	
57	0071	00000071	4200001622000000	
58	0072	00000072	4200001642000000	
59	0073	00000073	4200001662000000	
60	0074	00000074	4200001702000000	
61	0075	00000075	4200001722000000	
62	0076	00000076	4200001742000000	
63	0077	00000077	4200001762000000	
64	0100	00067000	4200002002000000	
65	0101	00000101	3676742052004000	
66	0102	00000102	4360702102000000	
67	0103	00000103	4200002632000000	
68	0104	00000104	0140702136002000	
69	0105	00000105	4355702156000000	
70	0106	00000106	4355702176000000	
71	0107	00000107	4355702216000000	
72	0110	00000110	4355702236000000	
73	0111	00000111	4355702256000000	
74	0112	00000112	4355702276000000	
75	0113	00000113	4355702316000000	GENERAL PURPOSE
76	0114	00000114	4355702336000000	MULTIPLY
77	0115	00000115	4355702356000000	MACRO INSTRUCTION
78	0116	00000116	4355702376000000	
79	0117	00000117	4355702416000000	
80	0120	00000120	4355702436000000	
81	0121	00000121	4355702456000000	
82	0122	00000122	4355702476000000	
83	0123	00000123	4355702516000000	
84	0124	00000124	4215702522000000	
85	0125	00000125	3664202542004000	
86	0126	00000126	4200000002000000	
87	0127	00000127	3200040002002000	
88	0130	00000130	4200000002000000	
89	0131	00000131	3360702642000000	
90	0132	00000132	0000202102002000	
91	0133	00000133	4200002662000000	
92	0134	00000134	4200002702000000	
93	0135	00000135	4200002722000000	
94	0136	00000136	4200002742000000	
95	0137	00000137	4200002762000000	
96	0140	00000140	4200003002000000	

TABLE 5. DETAILED EXECUTION OUTPUT

```
THIS INSTR AT:1425, M-FIELD = 1426 .
SUBROUTINE BOLN  ENTERED.
INSTR 04 EXECUTED, TEMP VALUE = 013404.
INSTR MOD = 3 JUST SET DATA-OUT = 00027010.
EXECUTED OP CODE 04 WITH MODIFIER 3.
A-REG 14 CONTENTS = 003001
B-REG 15 CONTENTS INITIALLY = 001400, NOW = 027010
Q-REG CONTENTS INITIALLY = 077777, NOW = 077777
IELEM CONTENTS INITIALLY = 001400, NOW = 016005

THE FOLLOWING CM INSTRUCTION HAS JUST BEEN EXECUTED:
1425     1074670542203400   04   03   14   15  M:1406   0 0 1 0 0 1 0 0 0 0 0 1 1 1 0 0 0 0 0 0

SCRATCH PAD(00) CONTAINS 001405 .
SCRATCH PAD(01) CONTAINS 000000 .
SCRATCH PAD(02) CONTAINS 000000 .
SCRATCH PAD(03) CONTAINS 000000 .
SCRATCH PAD(04) CONTAINS 000000 .
SCRATCH PAD(05) CONTAINS 000000 .
SCRATCH PAD(06) CONTAINS 000000 .
SCRATCH PAD(07) CONTAINS 000000 .
SCRATCH PAD(10) CONTAINS 004001 .
SCRATCH PAD(11) CONTAINS 016005 .
SCRATCH PAD(12) CONTAINS 000002 .
SCRATCH PAD(13) CONTAINS 000774 .
SCRATCH PAD(14) CONTAINS 003001 .
SCRATCH PAD(15) CONTAINS 027010 .
SCRATCH PAD(16) CONTAINS 077777 .
SCRATCH PAD(17) CONTAINS 000000 .

DATA IN = 00012004, DATA OUT = 00027010, BUSS = 00016005, C-REG = 00000005
CARRY = 1, SR1 = 0, SR2 = 0.
```

TABLE 6. FINAL OUTPUT

```
RUN TERMINATED AT TIME :    .10950 SECONDS AFTER    146 MICROS OR    34 MACROS EXECUTED.

SCRATCHPAD CONTAINS:

    00001442
    00000000
    00000000
    00000000
    00177303
    00176606
    00076605
    00000000
    00171547
    00161212
    00010512
    00177776
    00177777
    00002364
    00077777
    00035376

DATA IN = 00001777, DATA OUT = 00077777, BUSS = 00077777, C-REG = 1777, H-REG = 0000, Q-REG = 177777

A/D S =   00000000  00000000  00000000  00000000  00000000, D/A S =   00000000  00000000  00000000

 CARRY =   0, INPUT DISCRETES = 00000000

MSR1 = 0, MSR2 = 0
```

if the ability to vary more widely the intermediate detail will be worth the additional parameter testing that will be required to implement it, but further analysis is planned.

The structure of the control mechanisms of the emulator was of some concern, and it was not readily obvious what criteria should be used to manage a given emulation run. It was finally decided to use two logical (Boolean) variables, ERROR and FINIS, to control the runs. Both are initialized to .FALSE., and both are tested at the beginning of each simulated Macroinstruction execution. Any error condition in any subroutine causes an error message to be output at that point, and then causes the error variable to be set to .TRUE., which then causes the run to be terminated at the next macro-FETCH stage of execution. Upon completion of a macroinstruction, the normal run-termination conditions defined by the run control card are examined, and if the conditions are met, causes FINIS to be set .TRUE., with the same result. This technique allows the program termination to be controlled, and meets the structured programming criteria of one-entrance, one-exit for the main program. All subroutines have this same characteristic of one-in, out-out. In general, it is felt that valuable additional experience with the use of structured FORTRAN forms has been gained.

6. Conclusion and Future Work

The emulator described in this report is currently operational on the Raytheon R-520 and CDC 6600 computers, and is currently supporting the applications software development process for the first multi-purpose microprocessor feasibility application, that of guidance control computer on the T-6 missile. It is anticipated that extensions to the emulator will be made during the flight software development. Some of the planned extensions are:

a) Experimentation with various forms and frequencies of emulator output, and with interactive control of emulator runs.

b) Experimentation with different microprocessor software structures, such as the development of meaningful operation codes at the main memory level, as opposed to simply pointing to macros and data.

c) Experimentation with variations in the word size of main and control memory.

d) Efforts to implement the emulator on other host computers, including the HP 2100.

In addition to the above, but probably later in time, it is anticipated that studies will be made of the potential of generalizing from the current emulator form to allow external definition of instruction/control fields and their meanings. This will involve attempts to support microprogramming for other microprocessors, the accumulation of information on similarities and differences, and the analysis of this data

against the original structure of this emulation effort. Another planned extension will provide an interactive capability for testing and modifying microprograms.

One outstanding characteristic of this work that should be emphasised is the comparative ease of making necessary or experimental changes to the emulator. In a new-processor environment, where both unforeseen and planned hardware changes are frequent, the simplicity of making fast changes to the emulator program has been invaluable. The modular design coupled with the strict adherence to the principals of structured programming are primarily responsible for this capability. More specifically, when a hardware function was initially misunderstood, or later required modification, the exact location of the emulator representation of that function was immediately known, easily isolated, and readily changed. This feature is also expected to play a vital role in future experimentation with proposed hardware design variations, in that the effects as seen by software can be thoroughly evaluated very easily prior to making any actual hardware changes.

BIBLIOGRAPHY

Aitken, J. D., Sliz, C. J., and Leonard, J. P., T-6 Missile Description, US Army Missile Command, Redstone Arsenal, Ala., Technical Report RG-73-18, 10 August 1973.

Asquith, C. Frank, T-6 Digital Autopilot Data Processing Analysis and Specification, US Army Missile Command, Redstone Arsenal, Ala., Technical Report RG-75-36, 5 March 1973.

Copeland, E., Computer Evaluation Techniques, US Army Missile Command, Redstone Arsenal, Ala., Technical Report RG-72-3, January 1972.

Copeland, D. E., Jones, M. C., and Strickland, M. R., Improved HAWK Software Simulation, US Army Missile Command, Redstone Arsenal, Ala., Technical Report RG-75-1, 15 July 1974.

Appendix A. MULTIPURPOSE MICROPROCESSOR EMULATOR: DATA ELEMENT DEFINITIONS

1. Common

CNTMEM (1024	SIMULATED CONTROL MEMORY, 48 BITS/WORD
ICN	SIMULATED CARRY IN - 1 BIT (ACTUAL 24)
ICOUNT	INSTRUCTION COUNT REGISTER, INITIALIZED AT 0, INCREMENTED BY 1 IN EXECM (MICRO)
IDIN	SIMULATED DATA IN LINES - 16 BITS (ACTUAL 24)
IDOUT	SIMULATED DATA OUT LINES - 16 BITS (ACTUAL 24)
IDSCAR	SIMULATED INPUT DISCRETE REGISTER - 4 BITS (ACTUAL 24)
ITELM	SIMULATED TELEMETRY REGISTER - 16 BITS (ACTUAL 24)
MAINM (1024)	SIMULATED MAIN MEMORY, 16 BITS/WORD (ACTUAL 24)
MBUSS	SIMULATED DATA BUSS, 16 BITS (ACTUAL 24)
MEMC	SIMULATED MAIN MEMORY ADDRESS (C) REGISTER 10 BITS (ACTUAL 24)
MEMH	SIMULATED CONTROL MEMORY ADDRESS (H) REGISTER 10 BITS (ACTUAL 24)
MICAD (5)	SIMULATED A/D INPUT REGISTERS, 12 BITS EACH (ACTUAL 24)
MICDA (3)	SIMULATED D/A OUTPUT REGISTERS, 12 BITS EACH (ACTUAL 24)
MICQ	SIMULATED Q REGISTER - 16 BITS (ACTUAL 24)
MSCPAD (16)	SIMULATED MICROPROCESSOR SCRATCHPAD MEMORY - 16 BITS EACH (ACTUAL 24) MSCPAD (1) = PROGRAM COUNTER MSCPAD (2) THRU (4) = RETURN ADDRESSES MSCPAD (5) THRU (8) = X REGISTERS MSCPAD (9) THRU (12) = A REGISTERS MSCPAD (13) THRU (16) = B REGISTERS
MSR1	SIMULATED SR1
MSR2	SIMULATED SR2
NRUN (20)	RUN CONTROL PARAMETERS: NRUN (1) - PRESET VALUE FOR CONTROL MEMORY NRUN (2) - PRESET VALUE FOR MAIN MEMORY NRUN (3) - OUTPUT LEVEL NRUN (4) - INITIALIZATION OUTPUT LEVEL NRUN (5) - INITIAL PC VALUE (START) NRUN (6) - OUTPUT CONTROL INTERNAL - TIME NRUN (7) - OUTPUT CONTROL INTERVAL - INSTR. COUNT NRUN (8) - OUTPUT CONTROL - ADDRESS IN H NRUN (9) - TERMINATION CRITERIA NRUN (10) - TERMINATION VALUE NRUN (11) - FORMAT OF MEMORY LOAD USE REMAINING NRUN OPTIONS RESERVED FOR FUTURE USE -
FINIS	LOGICAL: NORMAL RUN TERMINATION FLAG
ERROR	LOGICAL: ERROR TERMINATION FLAG

MINSTR	MAIN MEMORY INSTRUCTION - 8 BITS (ACTUAL 24)
MOPND	MAIN MEMORY OPERAND - 8 BITS (ACTUAL 24)
MICRO (25)	MICROINSTRUCTION FIELDS: (ALL ACTUAL 24 BITS) MICRO (1) - OPERATION CODE - 5 BITS MICRO (2) - INSTRUCTION MODIFIER - 3 BITS MICRO (3) - OPERAND 1 ADDRESS - 4 BITS MICRO (4) - OPERAND 2 ADDRESS - 4 BITS MICRO (5) - NEXT MICRO (H) ADDRESS - 10 BITS MICRO (6) THU (24) - BINARY CONTROL FIELDS - 1 BIT EACH MICRO (25) - A/D MUX ADDRESS - 3 BITS
MCNT	MACRO COUNT - INCREMENTED IN MAIN
STIME	SIMULATED ELAPSED TIME - 750 NSEC/MICRO EXECUTION
ITIME	INTEGER VALUE OF STIME IN SECONDS
IETIM	ELAPEED EMULATION RUN TIME
KAD	KEY THAT D/A MUX HAS BEEN CLOCKED
KDA	KEY TO SELECTED A/D REGISTER

2. Noncommon

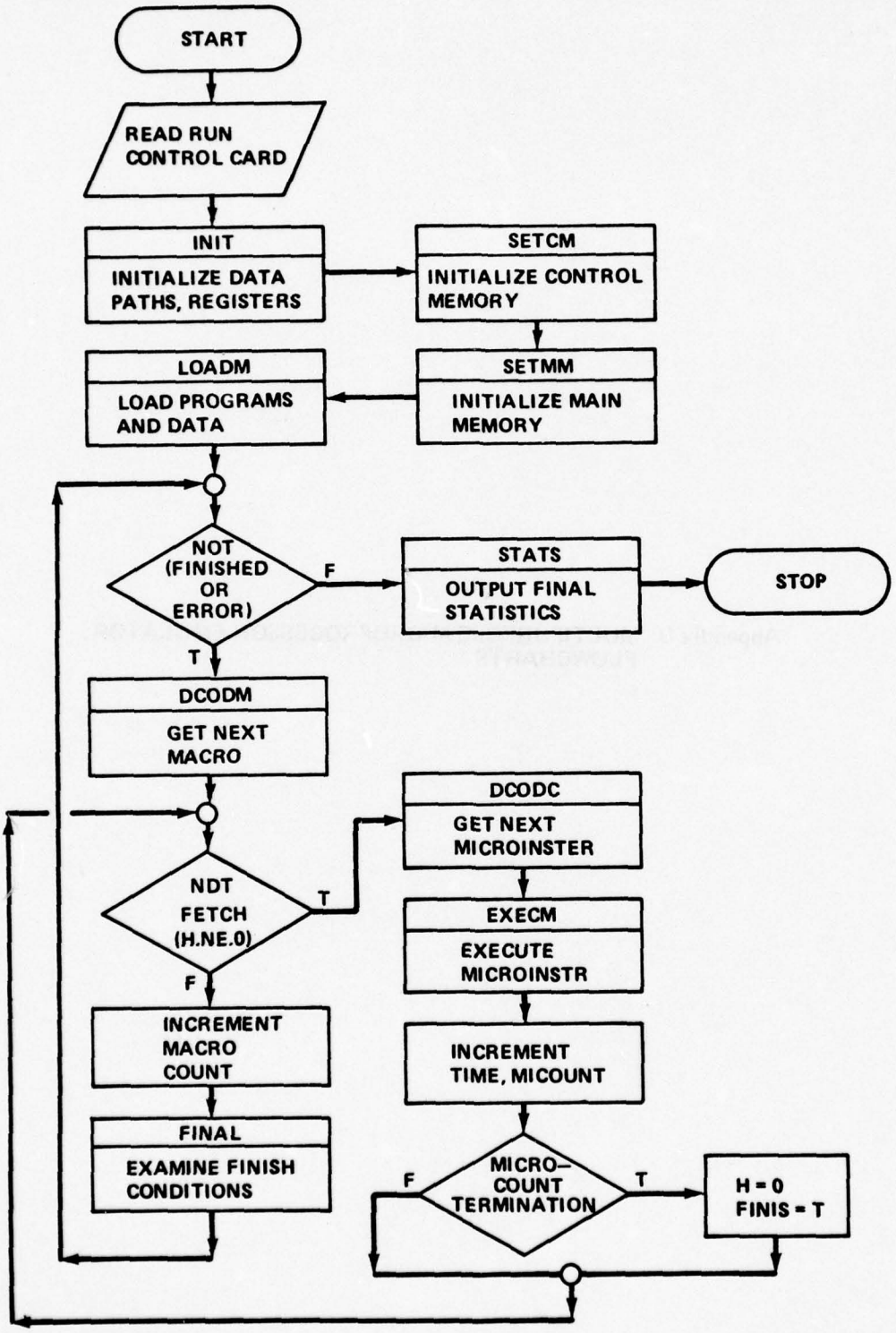
Subprogram	Parameter	
in EMUL (MAIN)	LFTMS	8-BIT MASK (377 OCTAL)
	TINCR	TIME INCREMENT (0.00000075 SEC = 75 NSEC)
	MICS	MICROINSTRUCTION COUNT
in SETCM	REPL /00/	TO PRESET CONTROL MEMORY TO CLEAR
	REPL 2	TO PRESET CM TO NOP TO HERE
	INDX	INDEX THRU MEMORY, 1 - 1024
	ADDR	INCREMENT H FIELD BY ONE (CDC VERSION ONLY)
	MSK	8-BIT MASK
SETMM	INDX	INDEX THRU MEMORY, 1 - 1024
LOADM	MSK 1	MASK FOR INVERSION OF BITS 2, 3 OF OP CODE
	KEY	MAIN OR CONTROL MEMORY, OR END LOAD
	MADR	OCTAL ADDRESS
	MAIN	VALUE FOR MAIN MEMORY
	CNTRL	VALUE FOR CONTROL MEMORY
	K	MAIN MEMORY WORD COUNTER
	L	CONTROL MEMORY WORD COUNTER
DCODC	IERR	CUMULATIVE ERROR FLAG
	IPTR	CALLING PARAMETER FOR MEMH
	MOP	CALLING PARAMETER FOR MICRO(N)
	INDX	MICRO INDEX
	NCOL	FIELD (COLUMN) INDEX
	IFLAG	ERROR RETURN FLAG
	MSK 1	MASK FOR INVERSION OF BITS 2, 3 OF OP CODE
DCODM	IOFF/01400	OFFSET FOR 1st INSTR OF EACH MACRO
	MIN	CALLING PARAMETER FOR MINSTR
	MOP	CALLING PARAMETER FOR MOPND

GETCM	IADR	ADDRESS OF SOURCE WORD
	IBITL	LEFT (START) BIT OF DESIRED FIELD
	IBITN	NUMBER OF BITS IN FIELD
	IBYTE	ADDRESS OF TARGET (FIELD) WORD
	IFLAG	ERROR FLAG
	IDR	IADR + 1 TO OFFSET 0 BASE FOR INDEX
	OPMSK/IMSK	MASK FOR EXTRACTING FIELD
	BYTE	48 - BIT ANSWER FIELD (R-520 VERSION ONLY)
	NMBR	FINAL SHIFT FOR RIGHT ADJUST
	ISTBIT	START BIT OFFSET FOR 60-BIT WORD (CDC VERSION ONLY)
GETMM	IADR	
	IBITL	
	IBITN	SAME AS GETCM
	IBYTE	
	IFLAG	
	IDR	
	IBIT	MACHINE-WORD ADJUSTMENT FOR START BIT
	IMASK	MASK FOR EXTRACTING FIELD
	NMBR	FINAL SHIFT FOR RIGHT ADJUST
INMOD	MOD	INSTRUCTION MODIFIER (MICRO(2)) - USED TO SPECIFY DESTINATION OF OUTPUT AND TO CONTROL SHIFT OPERATIONS.
	IA	INDEX OF A OPERAND
	IB	INDEX OF B OPERAND
	KMASK	RIGHTMOST 16-BIT MASK
	IHIBIT	LEFTMOST 1-BIT MASK (HIGH BIT)
XOR	IANS	ARGUMENT 1, RESULT OF EXCLUSIVE OR
(R-520 ONLY	J	ARGUMENT 2, VALUES TO BE EXLUSIVE
	K	ARGUMENT 3, OR ED
EXECM	IOP	OPERATION CODE (MICRO (1))
OPERATIONS	IOP	OP CODE (MICRO (1))
BOLN	LHIGH	OCTAL 177777 (FORCE ALL 1s), 16-BIT MASK
INVRT	MOD	INSTRUCTION MODIFIER (MICRO(2))
CMPLT	IAR	A OPERAND INDEX (MICRO(3))
TRINC	IBR	B OPERAND INDEX (MICRO(4))
DECTR		
SUM	ITMP	TEMPORARY HOLD FOR RESULTS OF EXECUTION UNTIL FINAL FORM/DESTINATIONS IS DETERMINED
DIFF		
PRBIN	ILOW	4 LSBs OF DATA IN (IDIN) OR DATA OUT (IDOUT)
BINRY	KOUNT	ELAPSED TIME (OPERATIONS) COUNTER FOR D/A OUTPUT REGISTER SELECTION
	MSK 10	10-BIT MASK FOR ADDRESSES
FLEX	K	KEY TO SELECTED DIRECT CODE OPTION

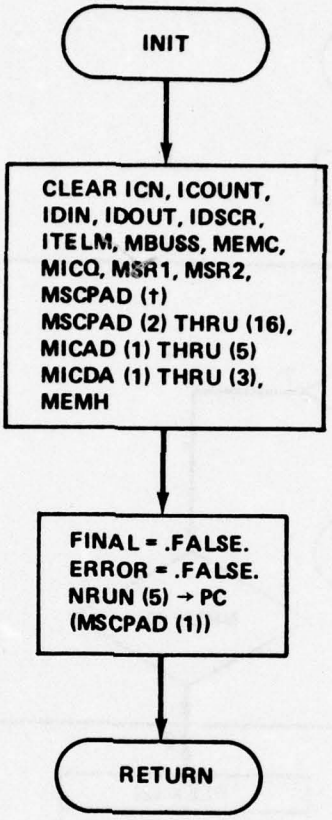
(R-520 ONLY)	FLTB	REAL (48-BIT) PARAMETER IN OR RETURN
	INTEGR	INTEGER (24-BIT) PARAMETER OR BOTH
FINAL	NEND	TERMINATION CRITERIA (NRUN(9))
	NVAL	TERMINATION VALUE (NRUN(10))
REPT1	ITEL2	PREVIOUS VALUE OF TELEMETRY REGISTER
	MICQ2	PREVIOUS VALUE OF Q-REGISTER

Appendix B. MULTIPURPOSE MICROPROCESSOR EMULATOR: FLOWCHARTS

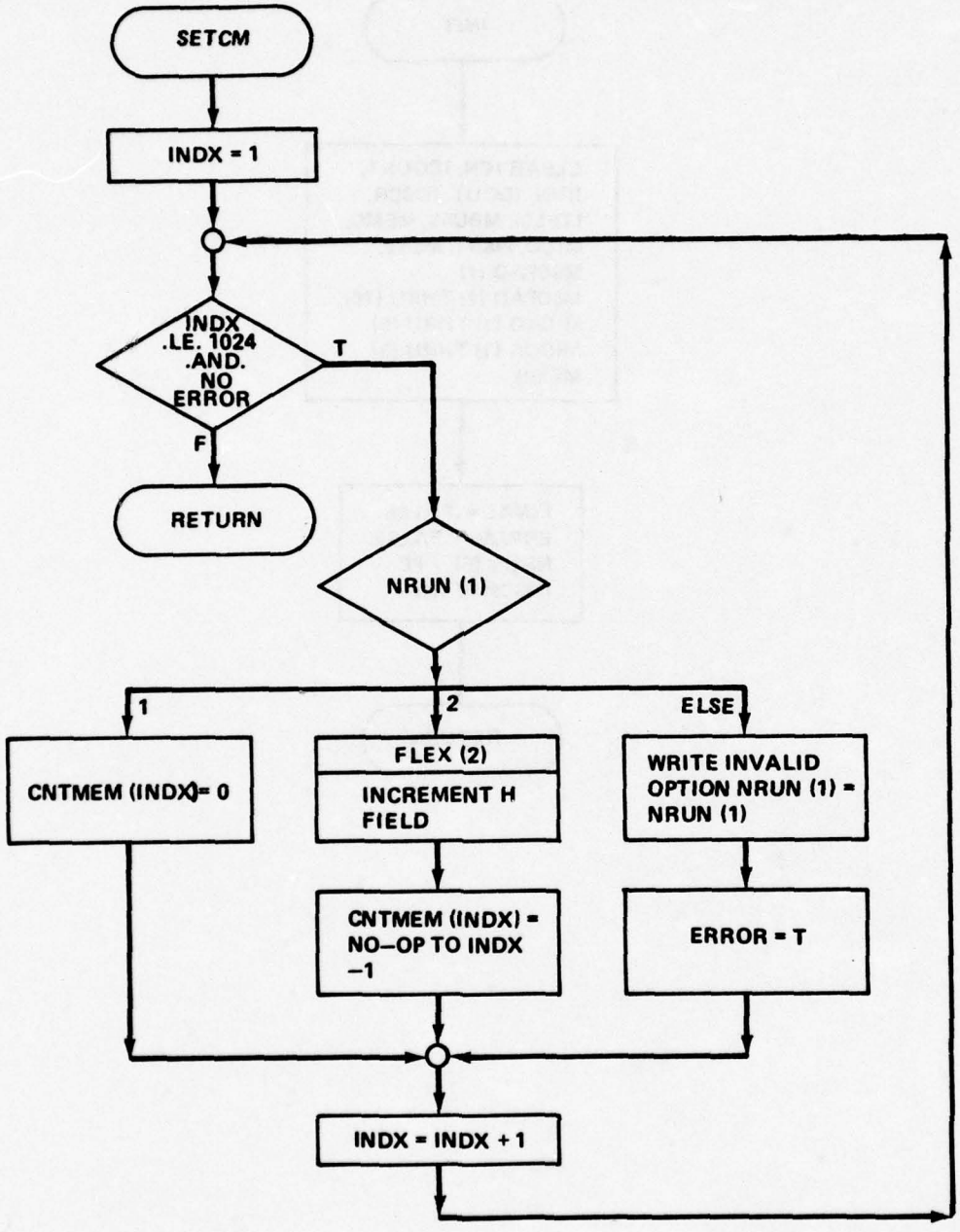
MULTIPURPOSE DIGITAL MICROPROCESSOR EMULATION MAIN PROGRAM



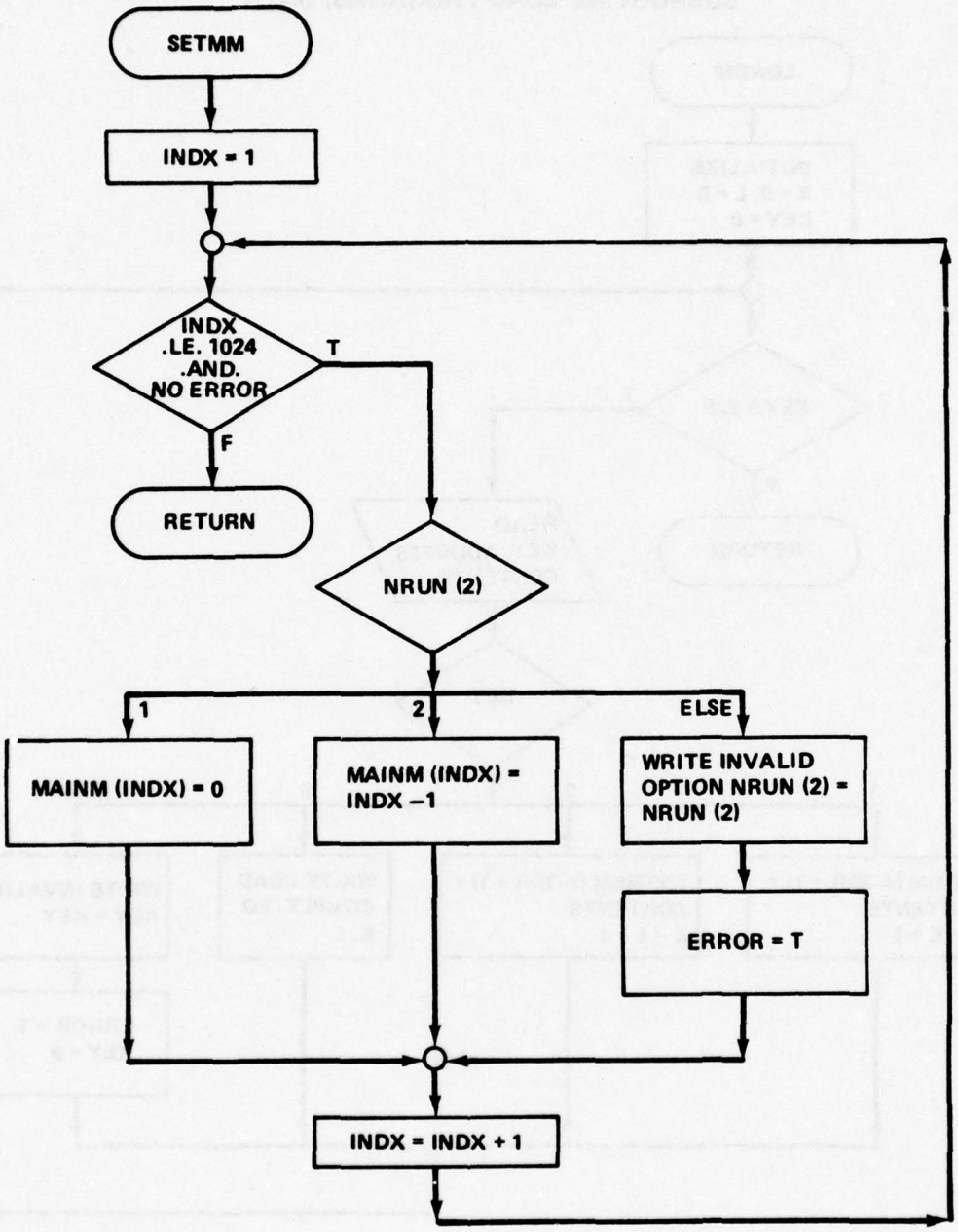
SUBROUTINE INITIALIZE (REGISTERS AND DATA PATHS)



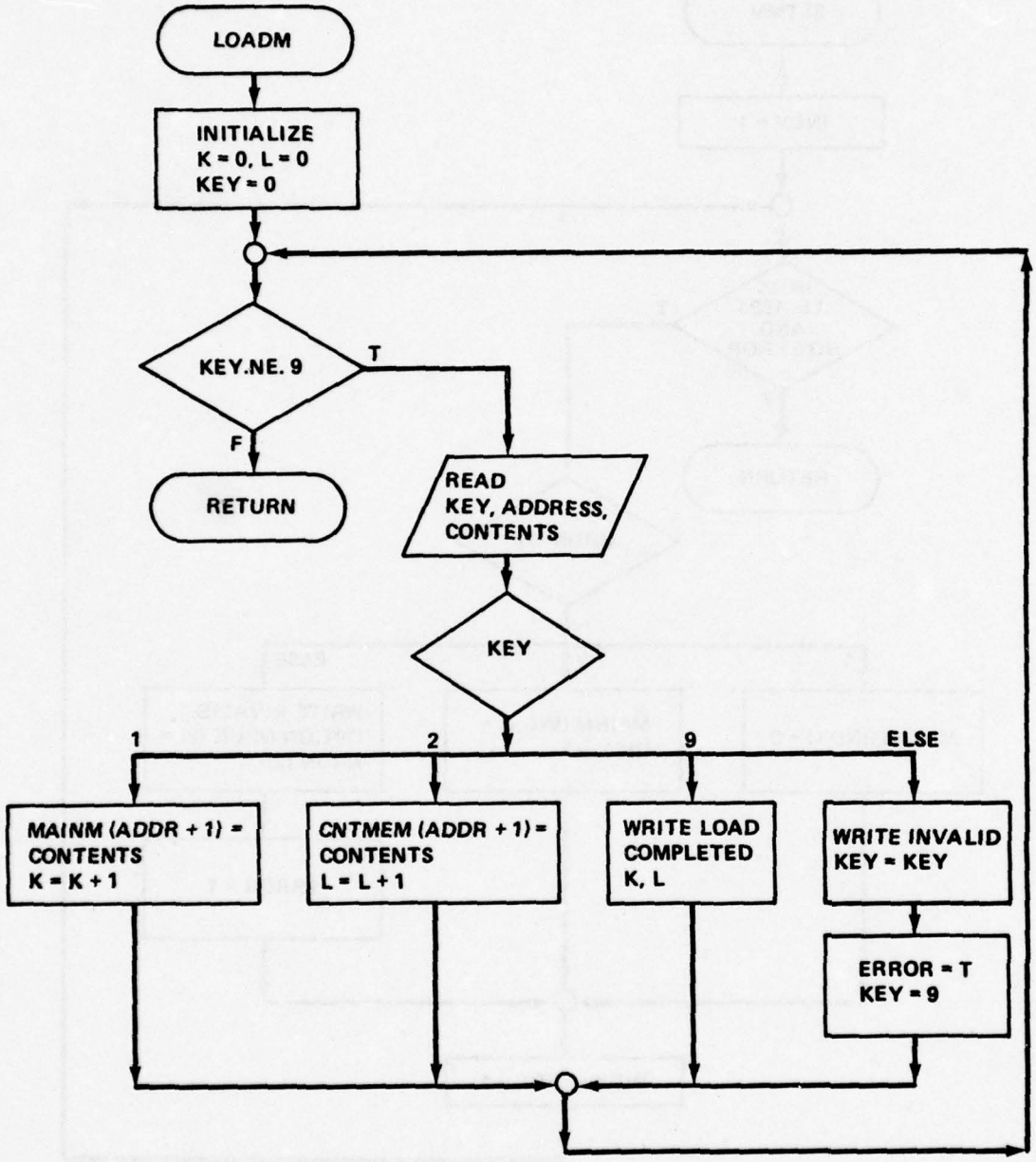
SUBROUTINE SET CONTROL MEMORY



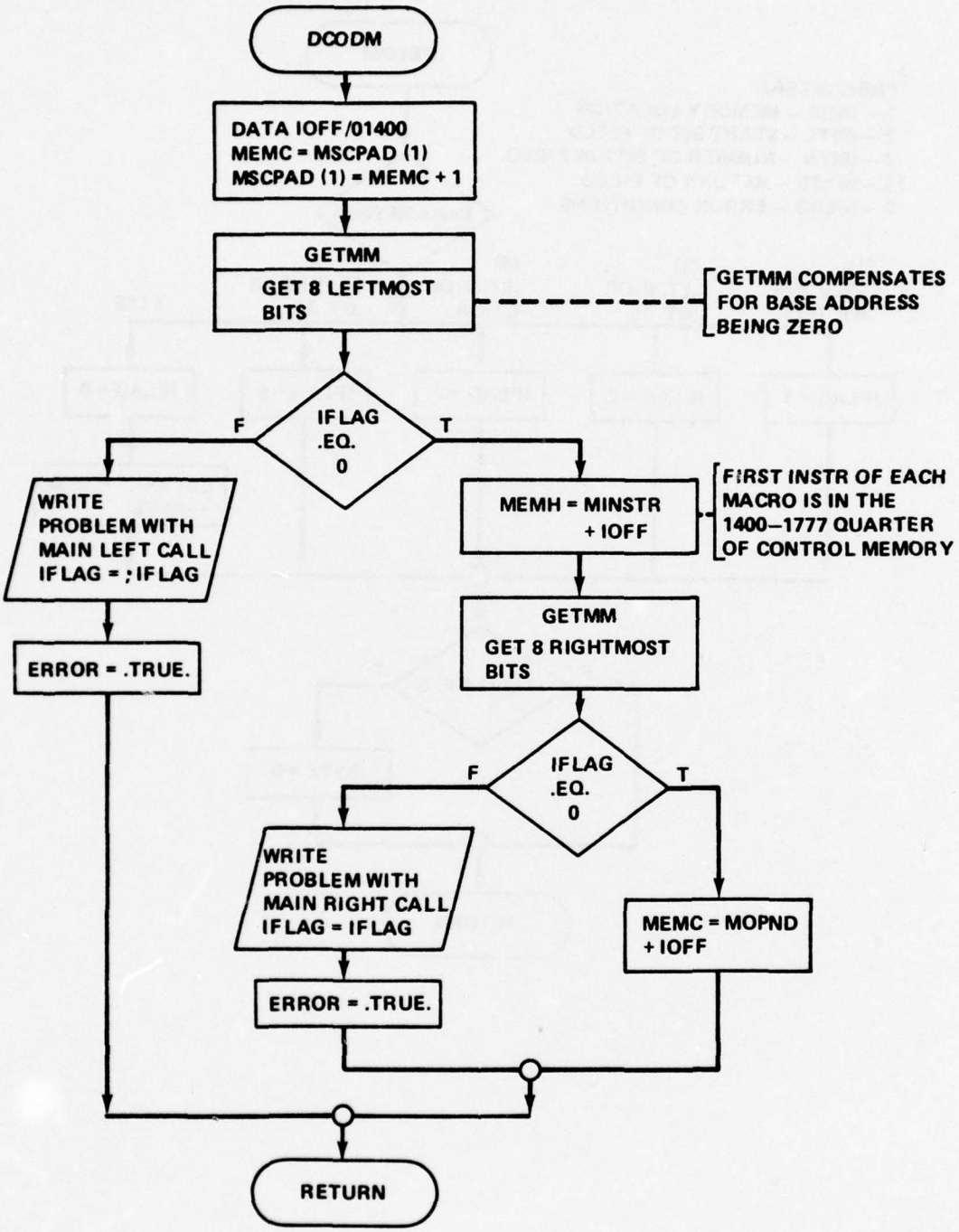
SUBROUTINE SET MAIN MEMORY



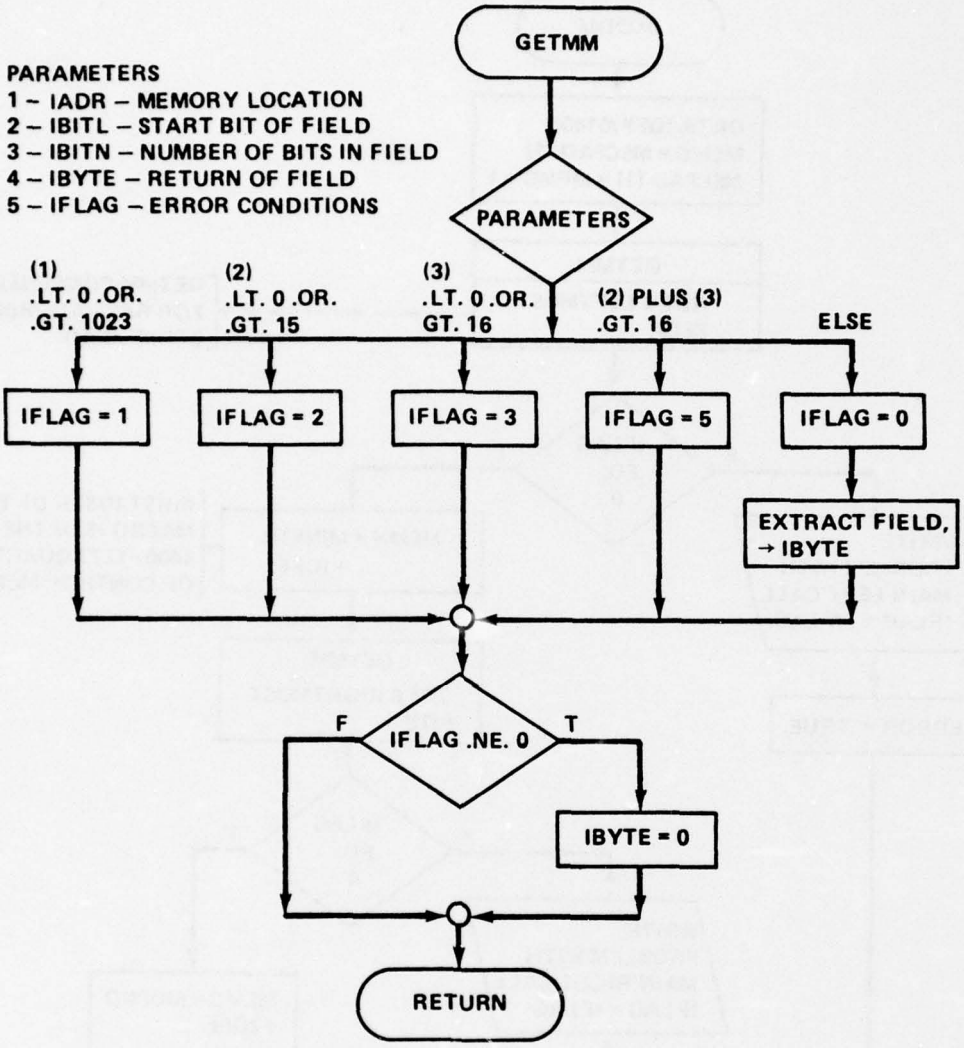
SUBROUTINE LOAD PROGRAMS, DATA



SUBROUTINE DECODE MAIN MEMORY WORD

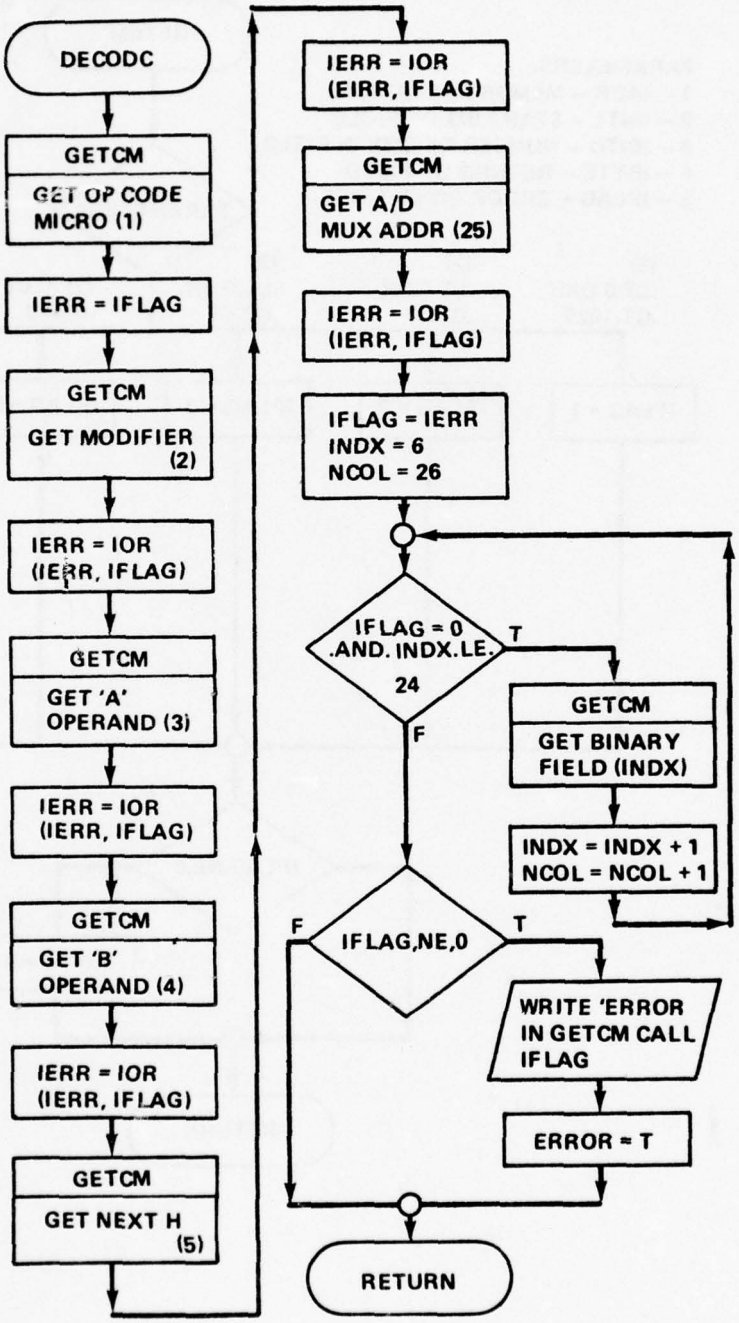


SUBROUTINE GET MAIN MEMORY FIELD

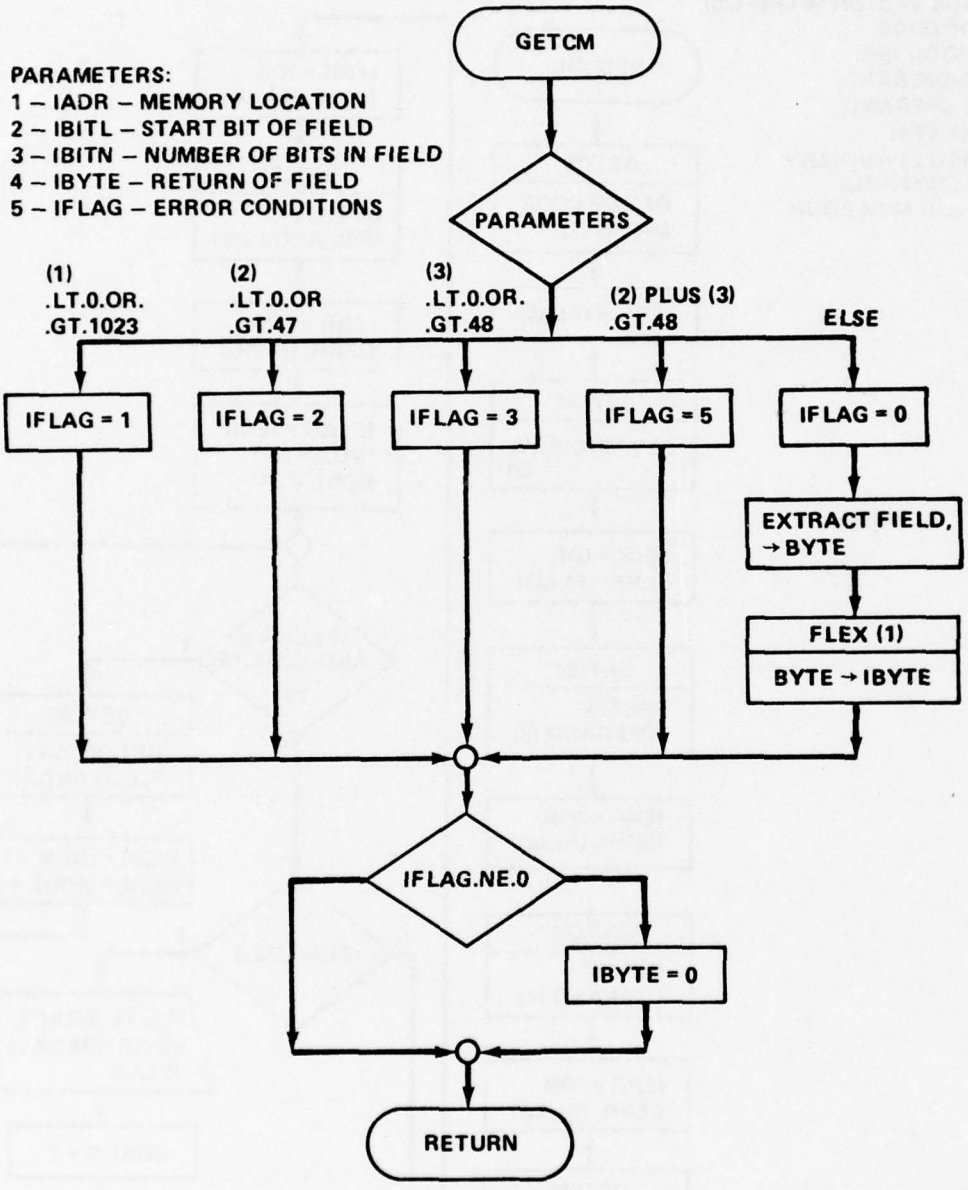


SUBROUTINE DECODE CONTROL MEMORY WORD

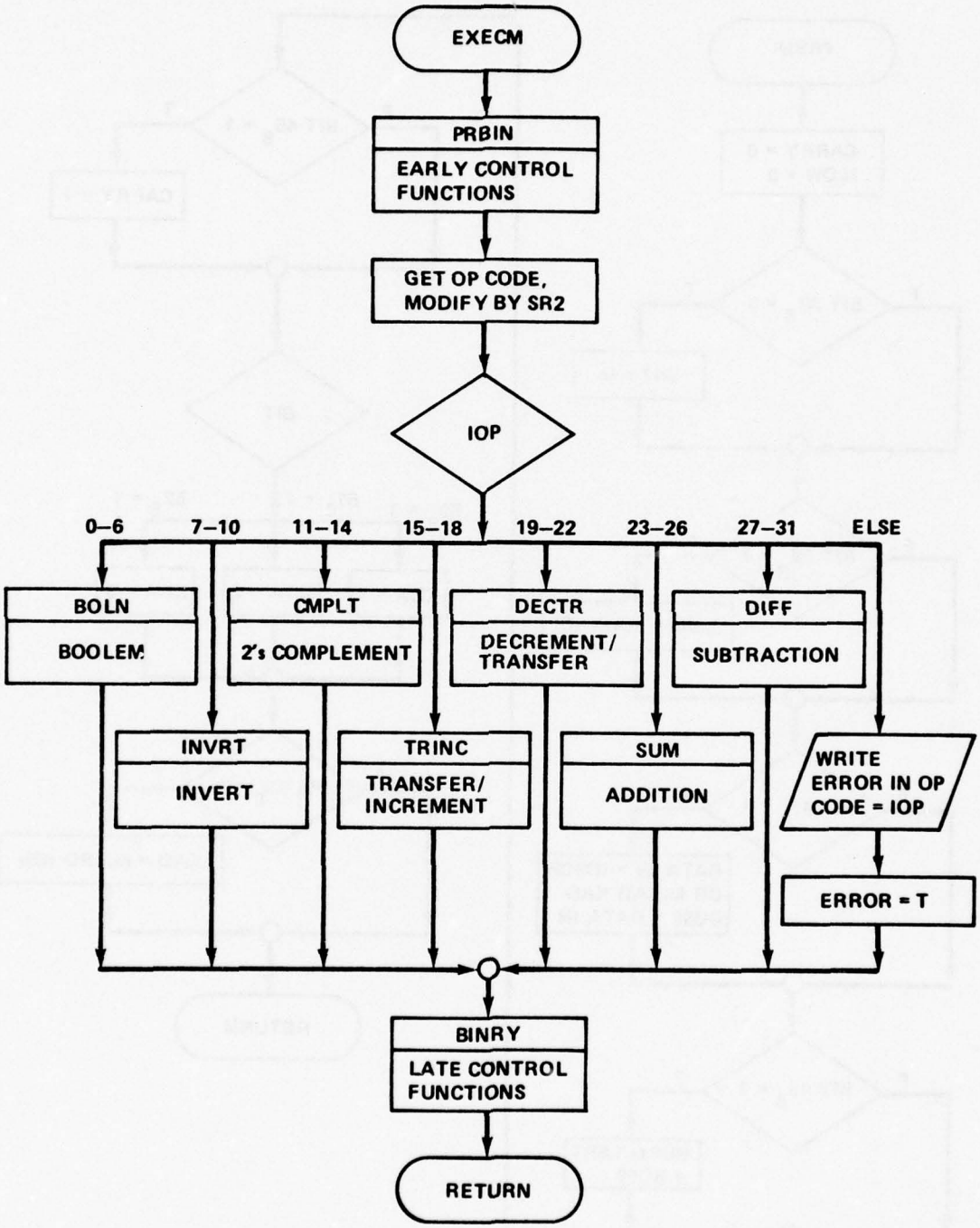
LOADS VECTOR MICRO (25)
1 = OP CODE
2 = MODIFIER
3 = A OPERAND
4 = B OPERAND
5 = NEXT H
6 THRU 24 = BINARY CONTROLS
25 = A/D MUX ADDR



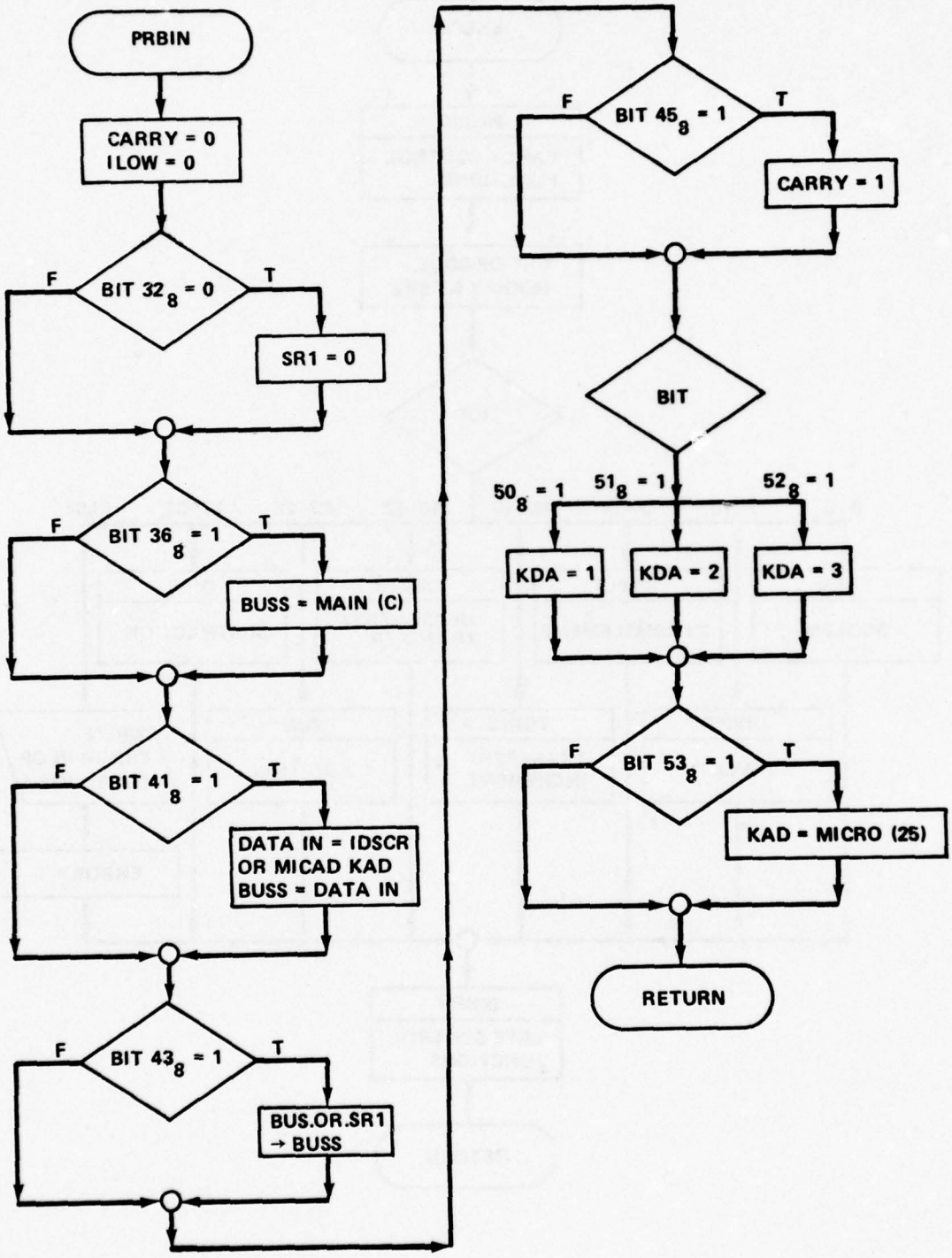
SUBROUTINE GET CONTROL MEMORY FIELD



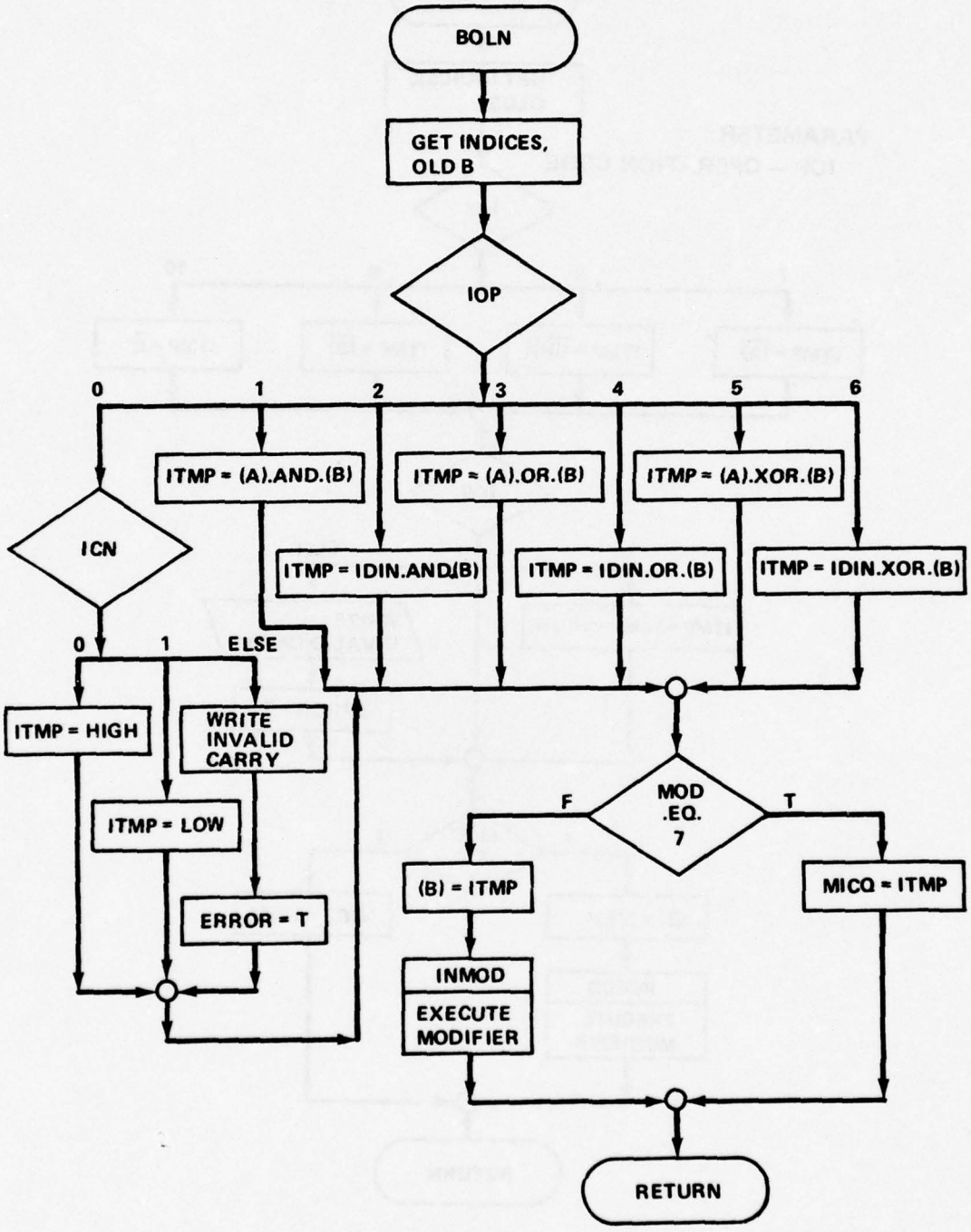
SUBROUTINE EXECUTE MICROISTRUCTION

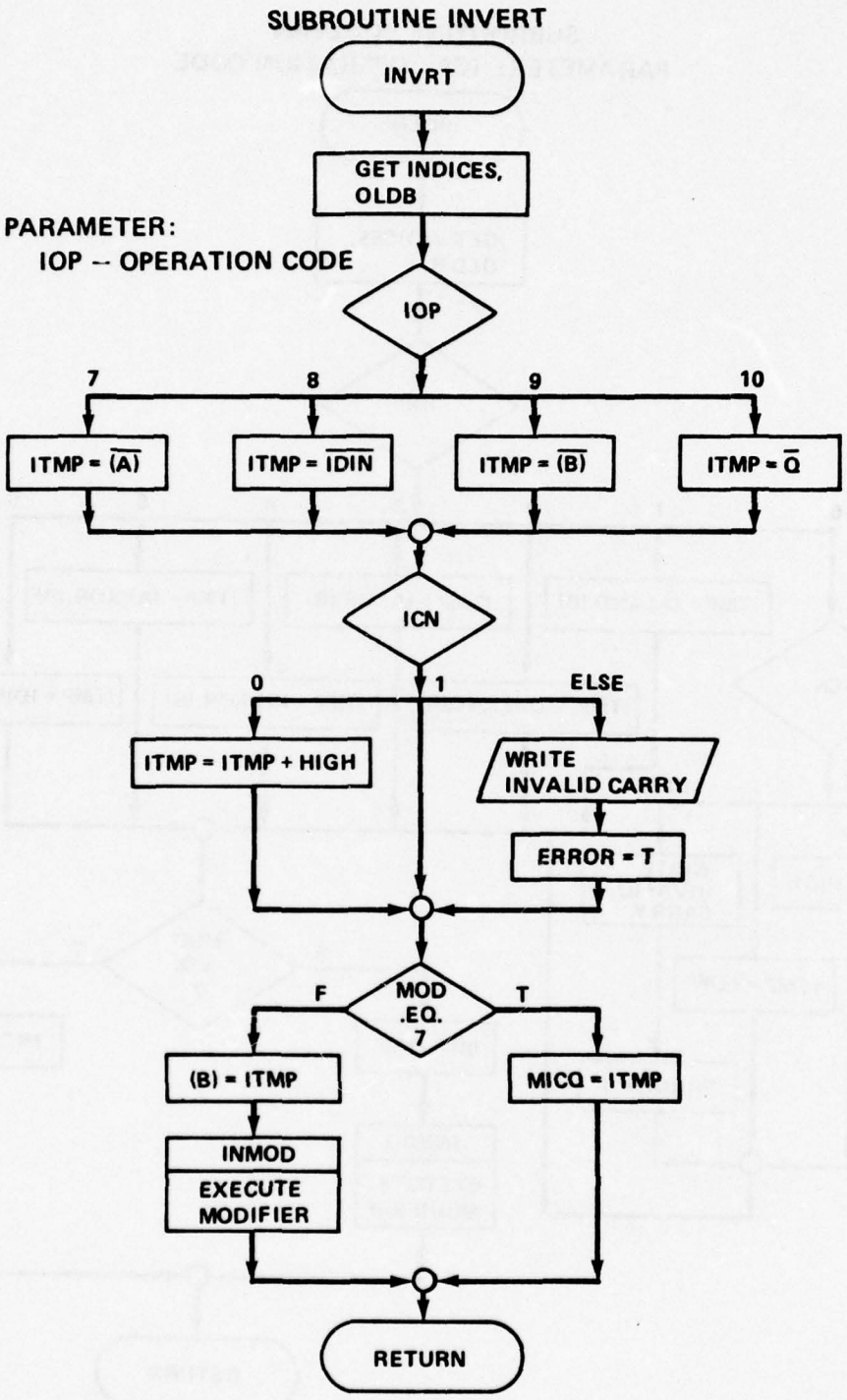


SUBROUTINE PRE–BINARY EARLY CONTROL FUNCTIONS
(BIT NUMBERS IN BINARY)

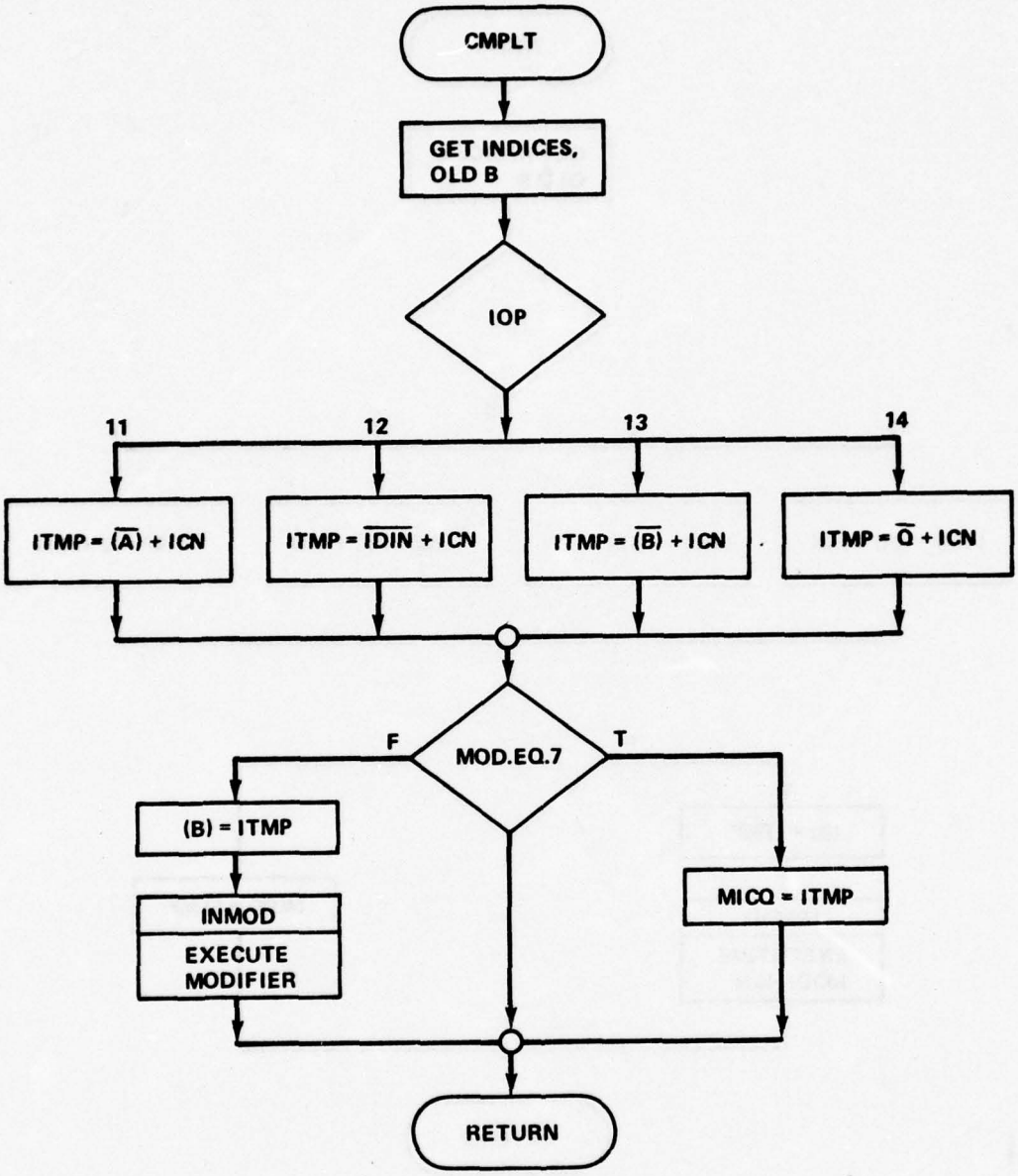


SUBROUTINE BOOLEAN
PARAMETER: IOP: OPERATION CODE

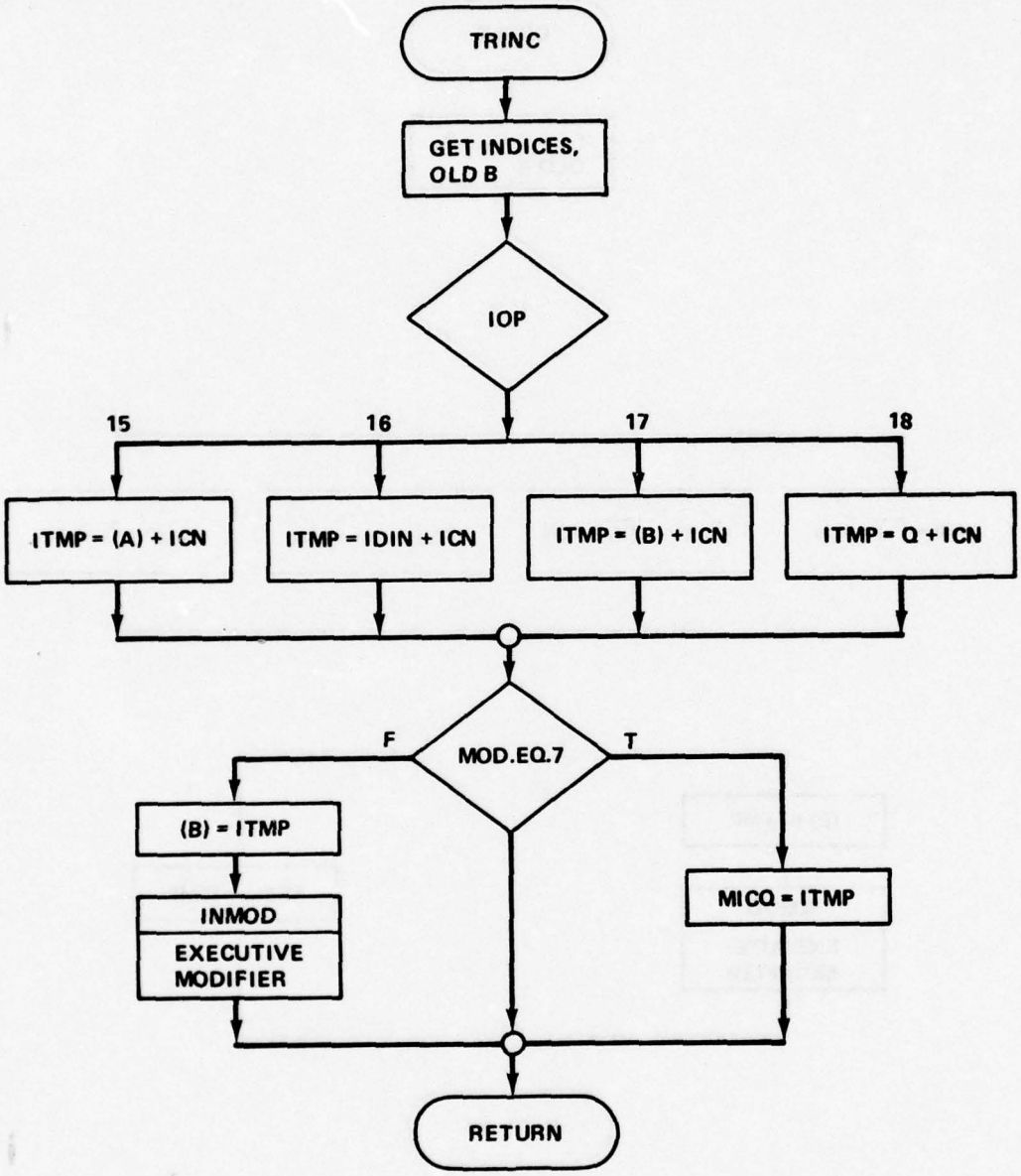



SUBROUTINE INVERT
INVRT
GET INDICES, OLDB
PARAMETER:
IOP -- OPERATION CODE
IOP
7
8
9
10
ITMP = (A)
ITMP = IDIN
ITMP = (B)
ITMP = Q
ICN
0
1
ELSE
ITMP = ITMP + HIGH
WRITE INVALID CARRY
ERROR = T
F
MOD .EQ. 7
T
(B) = ITMP
MICQ = ITMP
INMOD
EXECUTE MODIFIER
RETURN

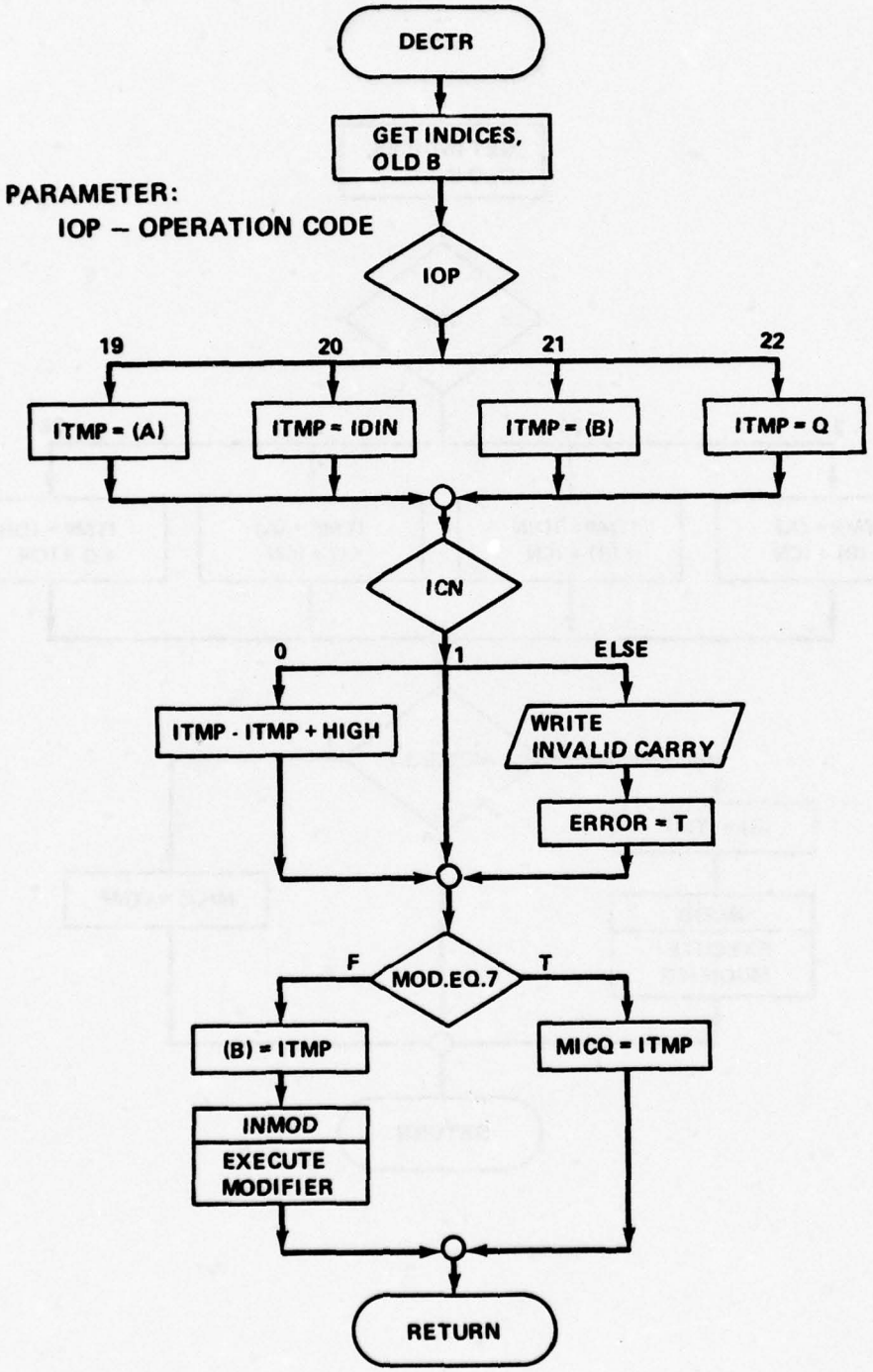
SUBROUTINE COMPLEMENT (2's) PARAMETER: IOP – OPERATION CODE



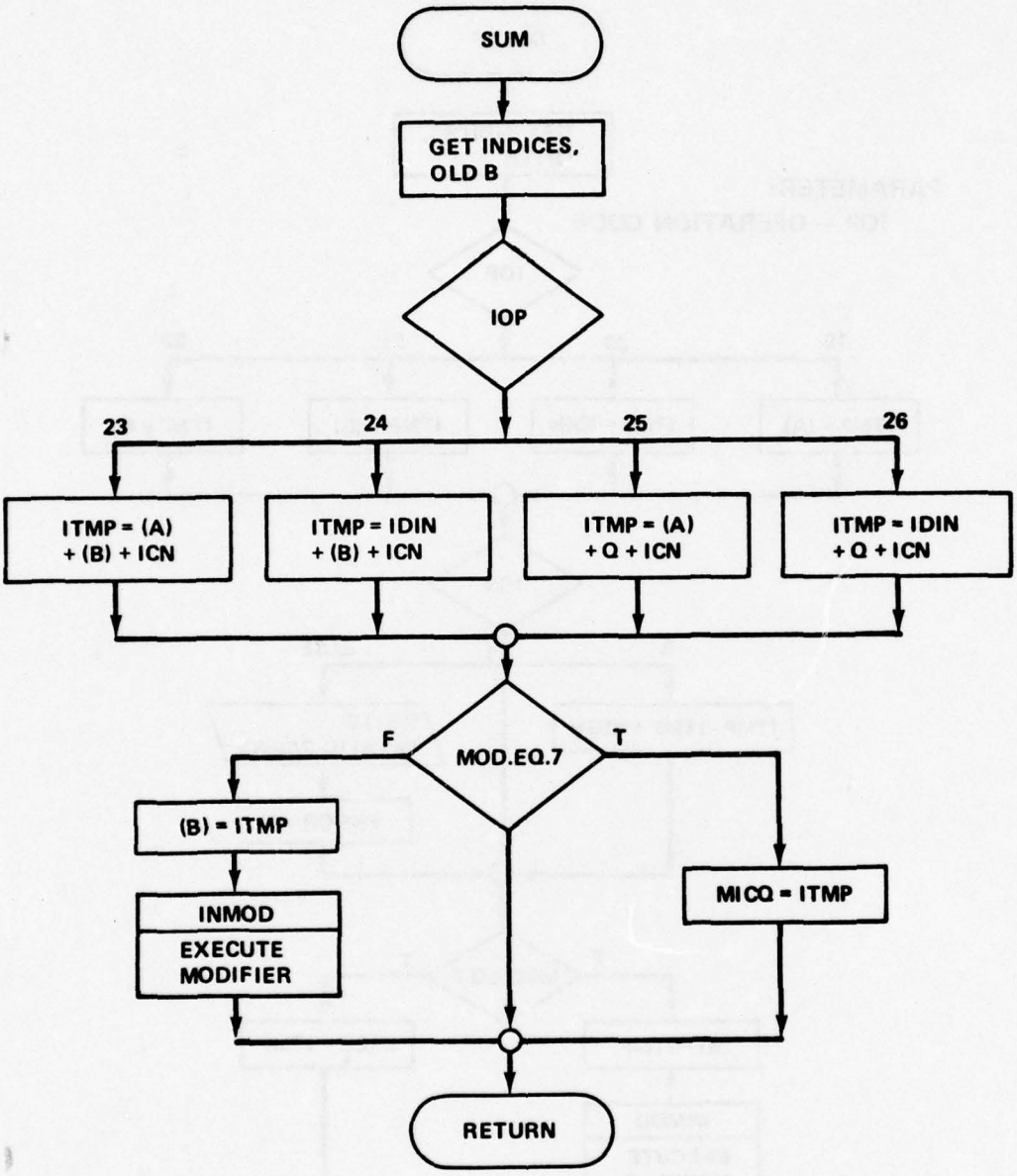
SUBROUTINE TRANSFER/INCREMENT PARAMETER: IOP – OPERATION CODE



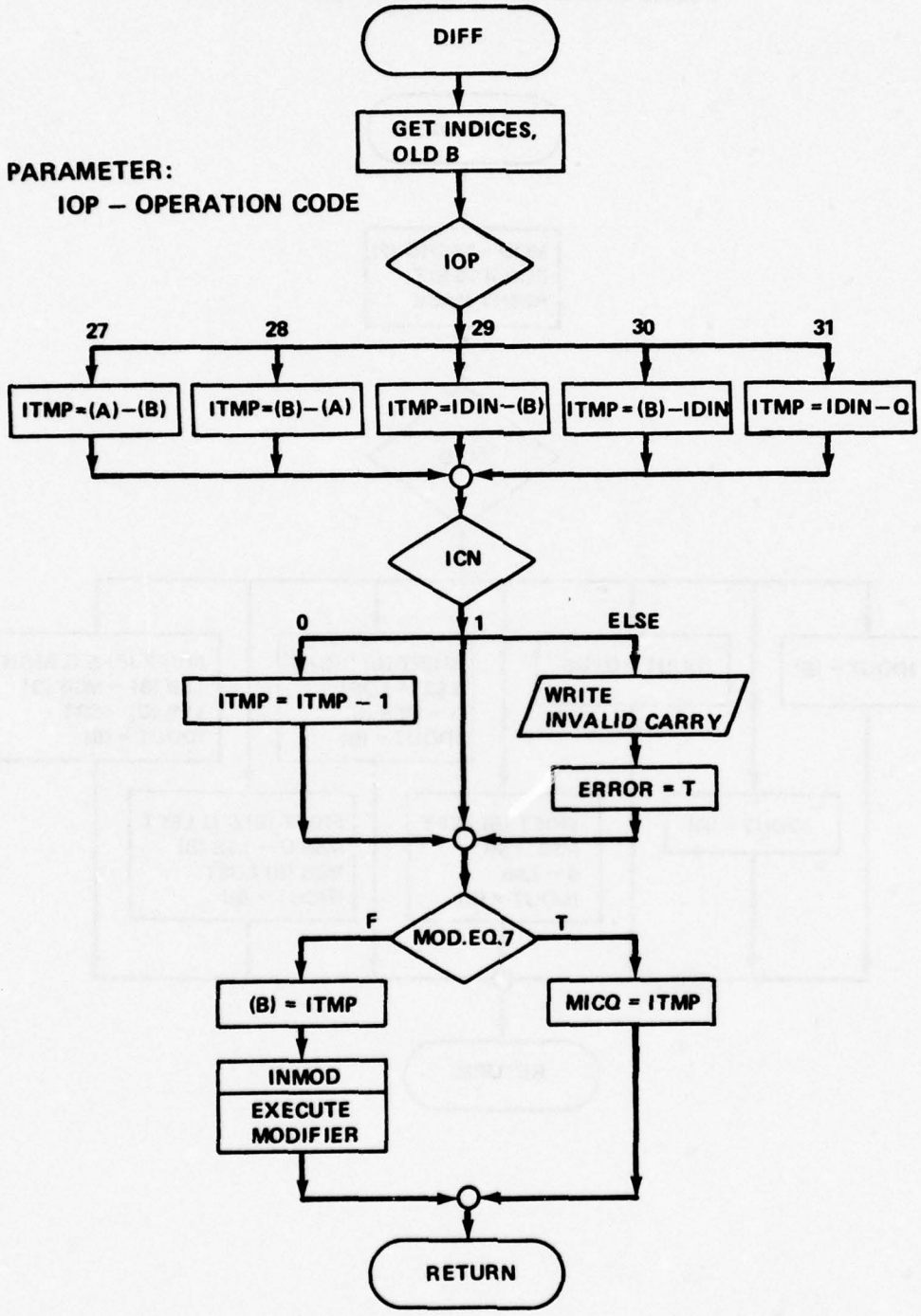
SUBROUTINE DECREMENT/TRANSFER



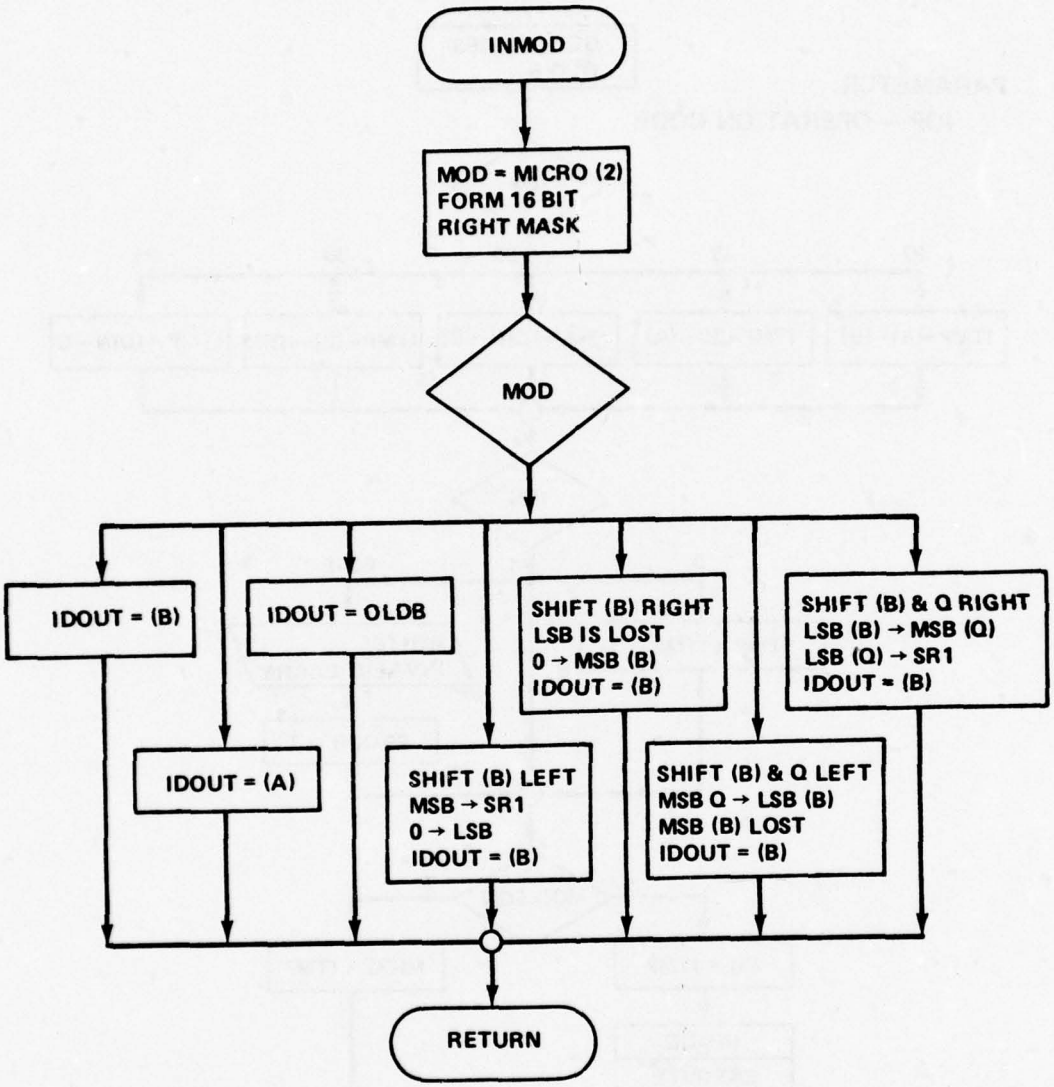
SUBROUTINE ADD PARAMETER: IOP – OPERATION CODE



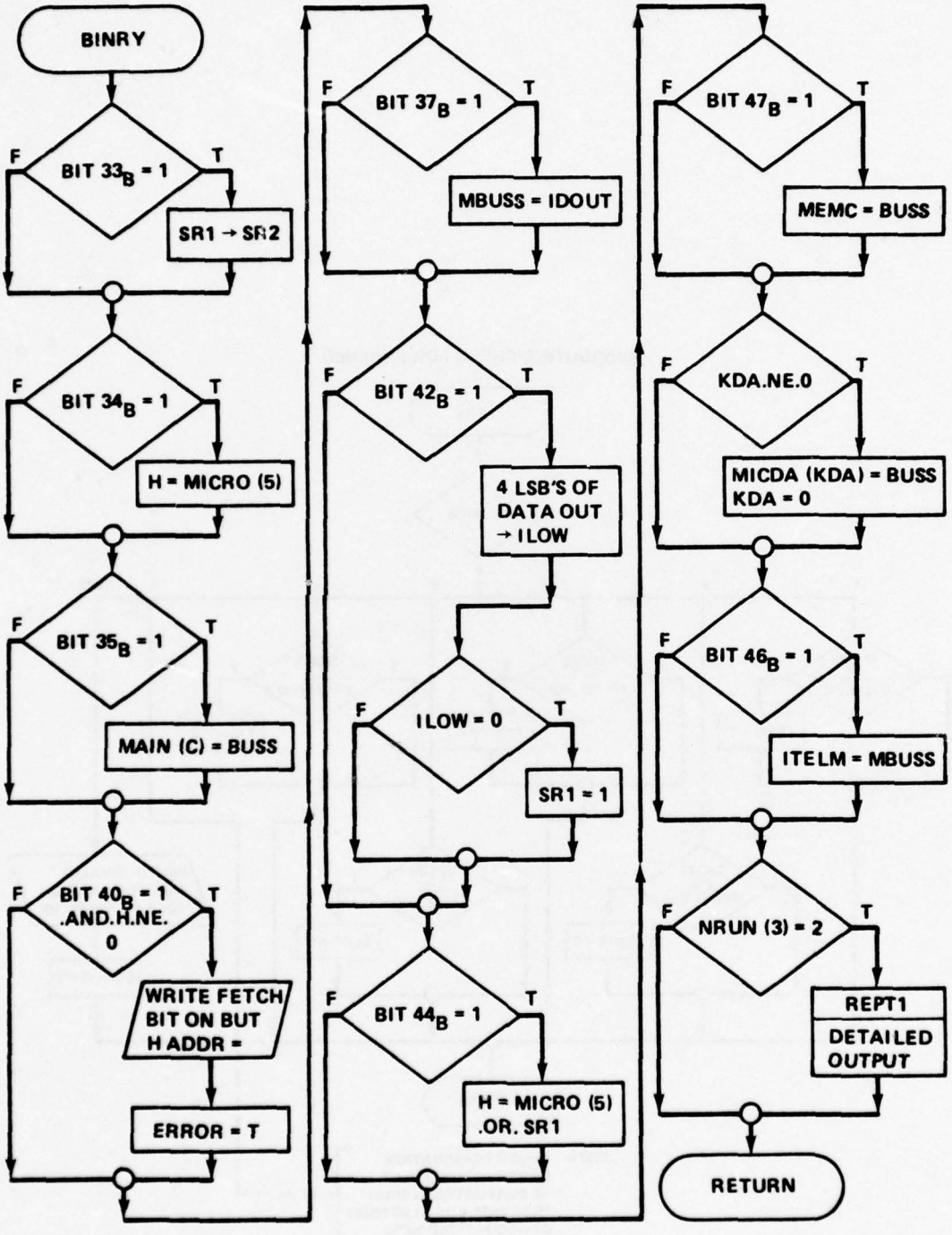
SUBROUTINE SUBTRACT



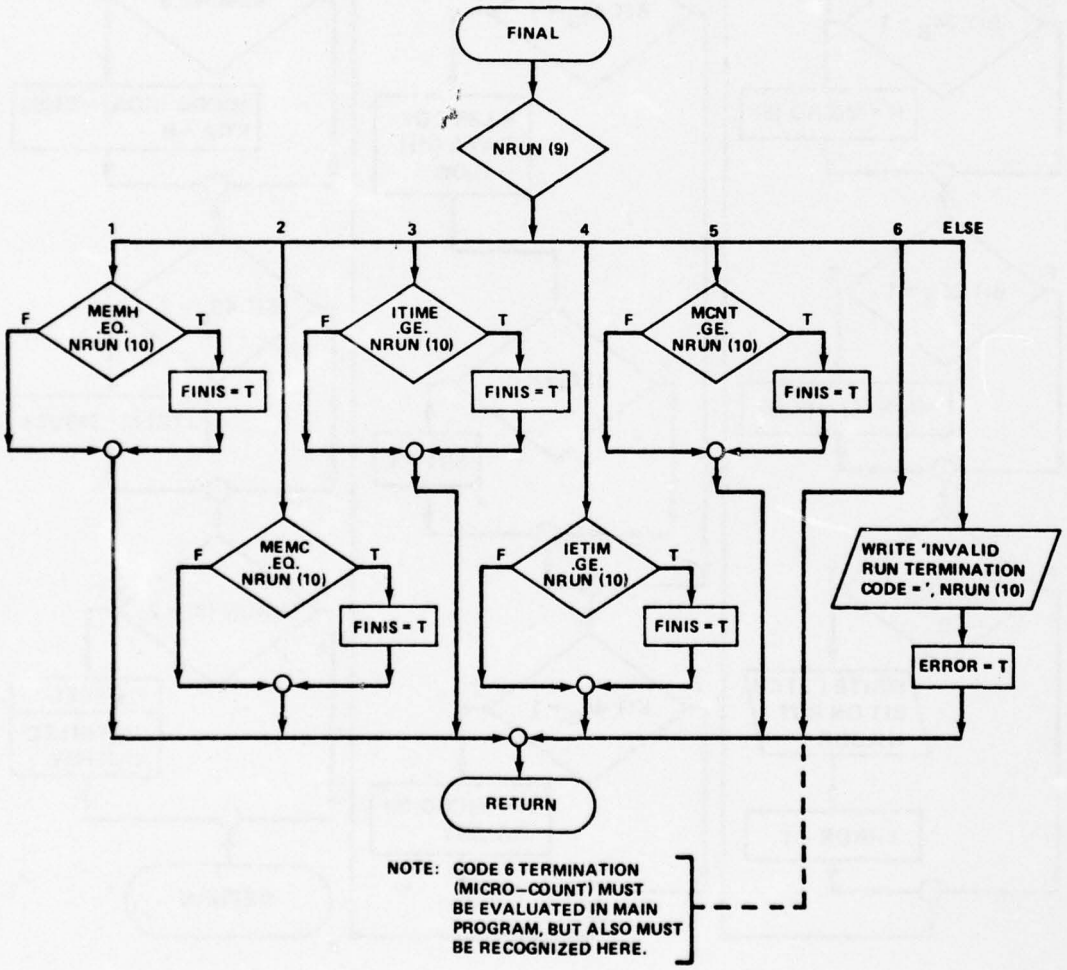
SUBROUTINE INSTRUCTION MODIFICATION



SUBROUTINE BINARY BIT NUMBERS IN OCTAL

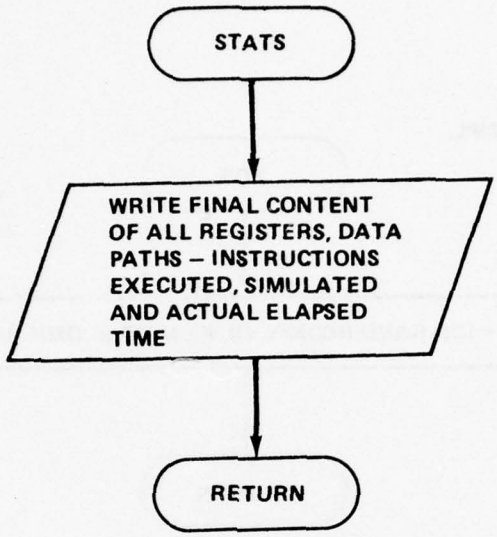


SUBROUTINE CHECK FOR FINISHED

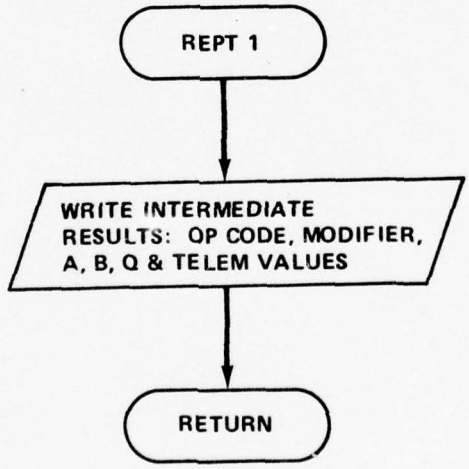


NOTE: CODE 6 TERMINATION (MICRO–COUNT) MUST BE EVALUATED IN MAIN PROGRAM, BUT ALSO MUST BE RECOGNIZED HERE.

SUBROUTINE FINAL STATISTICS

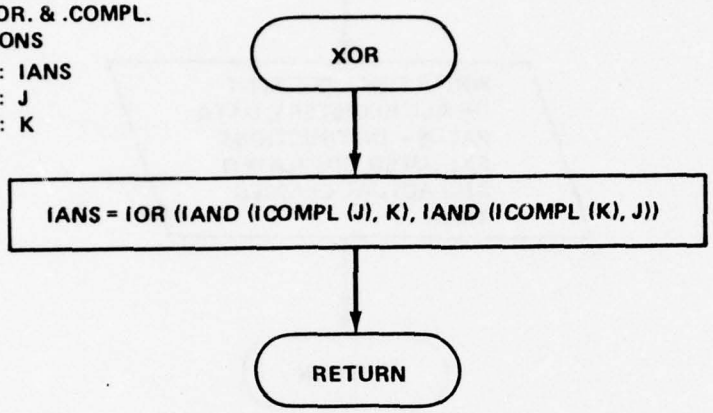


SUBROUTINE DIAGNOSTIC OUTPUT



SUBROUTINE EXCLUSIVE OR (RAYTHEON R520 VERSION ONLY)

USING FORTRAN
.AND., .OR. & .COMPL.
FUNCTIONS
ARG 1: IANS
ARG 2: J
ARG 3: K

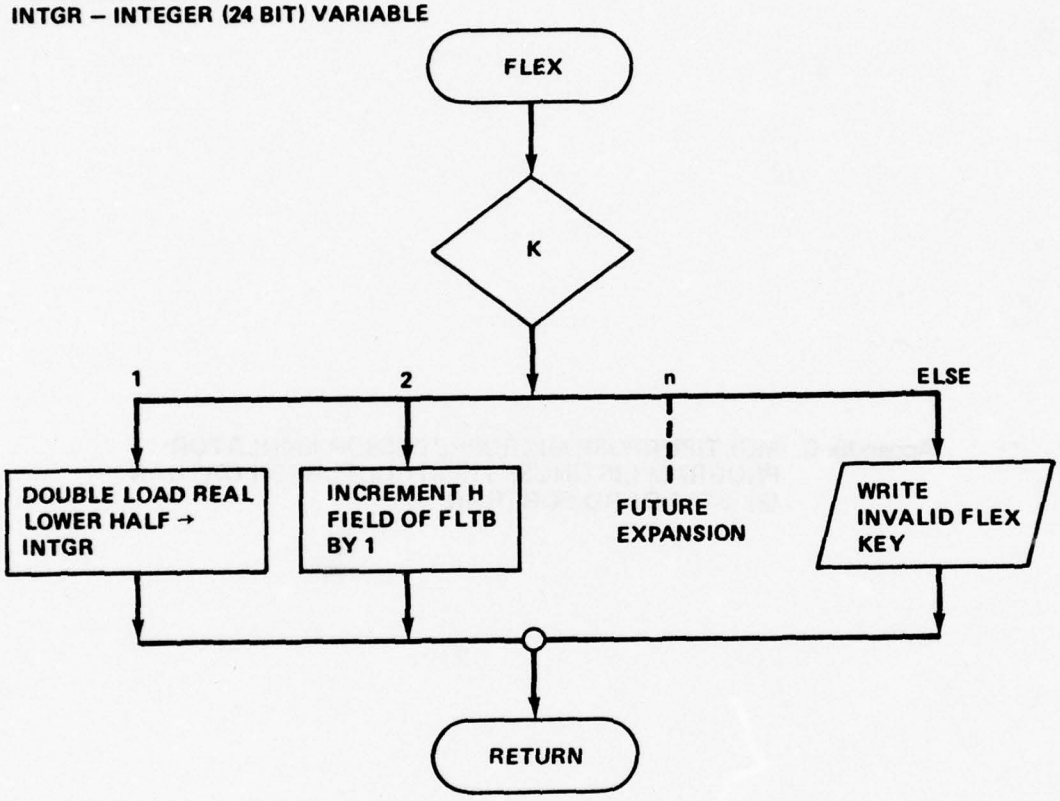


SUBROUTINE FLEXTRAN (RAYTHEON R520 VERSION ONLY)

DIRECT R–520 CODE

PARAMETERS:

K – KEY, = 1 FOR FLTB → INTGR
= 2 FOR INCREMENT H FIELD
FLTB – REAL (48 BIT) VARIABLE
INTGR – INTEGER (24 BIT) VARIABLE



Appendix C. MULTIPURPOSE MICROPROCESSOR EMULATOR: PROGRAM LISTINGS: (1) STRUCTURED FORTRAN (2) STANDARD FORTRAN

```
      PROGRAM EMUL (INPUT, OUTPUT, TAPE5 = INPUT, TAPE7=OUTPUT)          000110
C                                                                        000120
CCCCCC CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC 000130
C                                                                       *000140
C     PROGRAM NAME : MULTI-PURPOSE DIGITAL MICROPROCESSOR EMULATOR      *000150
C                    T - 5 AUTOPILOT VERSION                            *000160
C                                                                       *000170
C     AUTHOR : JERRY R BROOKSHIRE                                       *000180
C              US ARMY MISSILE COMMAND                                  *000190
C                                                                       *000200
C     PROJECT : MULTI-PURPOSE DIGITAL MICROPROCESSOR                    *000210
C                                                                       *000220
C     DESCRIPTION : THIS PROGRAM ESTABLISHES THE MICROPROCESSOR         *000230
C          HARDWARE ENVIRONMENT, PRESETS MAIN AND CONTROL MEMORY        *000240
C          AREAS, LOADS THE 48-BIT CONTROL MEMORY AND THE 16-BIT        *000250
C          MAIN MEMORY, RETRIEVES, DECODES AND EXECUTES CONTROL         *000260
C          MEMORY (MICRO-) INSTRUCTIONS UNDER MAIN MEMORY SEQUENCE      *000270
C          CONTROL, AND OUTPUTS COMPUTED VALUES PLUS PERFORMANCE        *000280
C          AND TIMING DATA UNDER RUN-OPTIONS AS SPECIFIED.              *000290
C                                                                       *000300
C     CALLING SEQUENCE : MAIN PROGRAM                                   *000310
C                                                                       *000320
C     SUBROUTINES CALLED :                                              *000330
C          SETCM - PRESET CONTROL MEMORY                                *000340
C          SETMM - PRESET MAIN MEMORY                                   *000350
C          INIT - INITIALIZE ALL REGISTERS AND DATA PATHS               *000360
C          LOADM - LOAD MAIN AND CONTROL MEMORIES FROM CARDS (TAPE)     *000370
C          DCODM - DECODE MAIN MEMORY , SET PC, M, RESET C REGS         *000380
C          DCODC - DECODE CONTROL (MICRO-) INSTRUCTION                  *000390
C          EXECM - EXECUTE MICRO INSTRUCTION                            *000400
C          FINAL - DETERMINE IF RUN IS COMPLETED OR ABORTED             *000410
C          STATS - OUTPUT FINAL STATISTICS UNDER OPTION CONTROL         *000420
C                                                                       *000430
C     VERSION  01  MOD  00     9 MAY 1975                               *000440
C                  MOD  01     6 JUNE 1975                              *000450
C                  MOD  02    25 JULY 1975                              *000460
C     VERSION  02  MOD  00    29 JULY 1976  (CDC 6600 VERSION)          *000470
C                  MOD  01    13 AUG  1975                             *000480
C                  MOD  02    19 SEP  1975                             *000490
C                  MOD  03     6 NOV  1975                             *000500
C                                                                       *000510
C***** ***************************************************************** 000520
C                                                                        000530
      LOGICAL FINIS, ERROR                                               000540
      COMMON    CNTMEM(1024)    , IGN    , ICOUNT , IOIN   ,           000550
     1          IOOUT  , IOSCR  , ITEL4  , MAINM(1024)     ,           000560
     2          MBUSS  , MEMC   , MEM4   , MICAD(6), MICDA(3),         000570
     3          MICQ   , MSCPAD(16)      , MSR1   , MSR2   ,           000580
     4          NRUN(20), FINIS , ERROR  , MINSTR , MOPND  ,           000590
     5          MICRO(25),MCNT  , STIME  , ITIME  , IETIM              000600
      COMMON    KAD    , KDA    , IOLD3                                 000610
      DATA LFTMS /3778/                                                  000620
C     GET RUN CONTROL PARAMETERS                                         000630
      READ (5, 10) NRUN                                                  000640
   10 FORMAT (20 0 4)                                                    000650
      WRITE (7, 20) NRUN                                                 000660
```

```
   20  FORMAT (20H1RUN CONTROL CARD =   , 2304 )                          000670
       TINCR = 0.00000075                                                 000680
       CALL INIT                                                          000690
       CALL SETCM                                                         000700
       CALL SETMM                                                         000710
       CALL LOADM                                                         000720
       MICS = 0                                                           000730
       IF (NRUN(1).EQ.1)                                                  000740
          THEN                                                            000750
C            PRINT MEMORY CONTENTS (PRE-EXECUTION):                       000760
             N = 1                                                        000770
             WRITE (7, 80)                                                000780
   30        FORMAT (1GH1 LOCATION,5X,15HMAIN   DECODED,26X,8HLOCATION,5X,000790
     .              15HMAIN   DECODED, /11H   DEC   OCT ,3X,12HMEMORY LEFT ,000800
     .               5HRIGHT,4X,14HCONTROL MEMORY,7X,9HDEC   OCT ,3X,     000810
     .              17HMEMORY LEFT RIGHT, 4X,14HCONTROL MEMORY //  )       000820
             DO WHILE (N.LE.1024)                                         000830
                NN = N - 1                                                000840
                NR = N + 1                                                000850
                MAIN = MAINM(N)                                           000860
                MAINL = SHIFT (MAIN, -8)                                  000870
                MAINR = AND (MAIN, LFTMS)                                 000880
                MAIN = MAINM(NR)                                          000890
                MAINJ = SHIFT(MAIN, -8)                                   000900
                MAINS = AND (MAIN, LFTMS)                                 000910
                WRITE (7, 90)NN,NN,MAINM(N),MAINL,MAINR,CNTMEM(N),        000920
     .                       N,N,MAINM(NR),MAINJ,MAINS,CNTMEM(NR)         000930
   90           FORMAT (1H ,I4,2X,O4,3X,O6,2X,O3,2X,O3,3X,O16,6X,         000940
     .                    I4,2X,O4,3X,O6,2X,O3,2X,O3,3X,O16    )          000950
                N = N + 2                                                 000960
             END DO                                                       000970
       END IF                                                             000980
C      MAJOR CONTROL LOOP (EXECUTE MACRO):                                000990
       DO WHILE (.NOT.(FINIS.OR.ERROR))                                   001000
          CALL DCODM                                                      001010
C         MINOR CONTROL LOOP (EXECUTE MICRO):                           001020
          DO WHILE (MEMM.NE.0)                                            001030
             MMOLD = MEMM                                                 001040
             CALL DCODC                                                   001050
             CALL EXECM                                                   001060
             IF (NRUN(3).EQ.2)                                            001070
                THEN                                                      001080
C                  OUTPUT EXECUTION DETAILS:                            001090
                   N = MMOLD + 1                                          001100
                   WRITE (7, 15) MMOLD, CNTMEM(N), (MICRO(I), I = 1, 25)  001110
   15              FORMAT (53H0THE FOLLOWING CM INSTRUCTION HAS JUST BEEN EXECUTED: ,001120
     .                    //1H , O4, 5X,O18,4(3X,O2),4H  M: ,O4,2X, 20(1X,I1)///)  001130
                   M = 1                                                  001140
                   DO WHILE (M.LE.16)                                     001150
                      K = M - 1                                           001160
                      WRITE (7, 30) K, MSCPAD(M)                          001170
   30                 FORMAT (13H SCRATCH PAD( , O2, 11H) CONTAINS  , O6,  2H .)  001180
                      M = M + 1                                           001190
                   END DO                                                 001200
                   WRITE (7, 50) IDIN, IDOUT, MBUSS, MEMC, ICN, MSR1, MSR2  001210
   50              FORMAT (11HGDATA IN = ,O8,13H, DATA OUT = ,O8,9H, BUSS = ,O8,  001220
```

```
     .                  10H, C-REG = ,O8,/9H CARRY = ,I1,8H, SR1 = ,I1,8H, SR2 = ,O01230   001240
     .                  I1, 1H.)                                                          001250
           END IF                                                                        001260
C          INCREMENT TIME AND MICRO COUNT:                                              001270
           STIME = STIME + TINCR                                                        001280
           ITIME = IFIX(STIME)                                                          001290
           MICS = MICS + 1                                                              001300
           IF (NRUN(3).EQ.2)                                                            001310
              THEN                                                                      001320
                 WRITE (7, 68) STIME, ITIME, MICS                                       001330
  +3             FORMAT (1H0, F15.8, 5X, I8, 5X, I5)                                   001340
           END IF                                                                       001350
C          CHECK FOR MICRO-COUNT TERMINATION:                                           001360
           IF (NRUN(9).EQ.0)                                                            001370
              THEN                                                                      001380
                 IF (MICS.GE.NRUN(10))                                                  001390
                    THEN                                                                001400
                       MEMM = 0                                                         001410
                       FINIS = .TRUE.                                                   001420
                 END IF                                                                 001430
           END IF                                                                       001440
        END DO                                                                          001450
C       INCREMENT MACRO COUNT:                                                          001460
        MCNT = MCNT + 1                                                                 001470
        CALL FINAL                                                                      001480
     END DO                                                                             001490
     IF (NRUN(4).EQ.1)                                                                  001500
        THEN                                                                            001510
C          PRINT MEMORY CONTENTS (POST-EXECUTION):                                      001520
           N = 1                                                                        001530
           WRITE (7,80)                                                                 001540
           DO WHILE (N.LE.1024)                                                         001550
              NN = N - 1                                                                001560
              NR = N + 1                                                                001570
              MAIN = MAINM(N)                                                           001580
              MAINL = SHIFT (MAIN, -8)                                                  001590
              MAINR = AND (MAIN, LFTMS)                                                 001600
              MAIN = MAINM(NR)                                                          001610
              MAINJ = SHIFT (MAIN, -8)                                                  001620
              MAINS = AND (MAIN, LFTMS)                                                 001630
              WRITE (7, 90) NN,NN,MAINM(N),MAINL,MAINR,CNTMEM(N),                       001640
     .                      NR,NR,MAINM(NR),MAINJ,MAINS,CNTMEM(NR)                      001650
     .         N = N + 2                                                              001660
           END DO                                                                       001670
     END IF                                                                             001680
     CALL STATS                                                                         001690
     STOP                                                                               001700
     END

     0 ERRORS FOUND IN THIS ROUTINE.   19 CARDS READ.          130 CARDS OUTPUT.
```

```
      SUBROUTINE INIT                                                   001710
C                                                                       001720
C***** ***************************************************************001730
C                                                                      *001740
C     ONE TIME INITIALIZATION OF ALL REGISTER  AND DATA PATHS          *001750
C     INCLUDING SETTING OF PC AND M REGISTERS FROM RUN CONTROL CARD    *001760
C                                                                      *001770
C***** ***************************************************************001780
C                                                                       001790
      LOGICAL   FINIS  , ERROR                                          001800
      COMMON    CNTMEM(1024)     , IGN    , ICOUNT , IOIN   ,          001810
     1          IDOUT  , IDSCR   , ITELM  , MAINM(1024)     ,          001820
     2          MBUSS  , MEMC    , MEMM   , MICAD(6), MICOA(3),         001830
     3          MICQ   , MSCPAD(16)       , MSR1   , MSR2   ,          001840
     4          NRUN(20), FINIS  , ERROR  , MINSTR , MOPND  ,          001850
     5          MICRO(25),MCNT   , STIME  , ITIME  , IETIM             001860
      COMMON    KAD    , KDA    , IOLD3                                 001870
      IGN = 3                                                           001880
      ICOUNT = 0                                                        001890
      IOIN = 0                                                          001900
      IDOUT = 0                                                         001910
      IDSCR = 0                                                         001920
      ITELM = 0                                                         001930
      KAD = 0                                                           001940
      KDA = 3                                                           001950
      MBUSS = 0                                                         001960
      MEMC = 0                                                          001970
      MEMM = 0                                                          001980
      MICQ = 0                                                          001990
      MSR1 = 0                                                          002000
      MSR2 = 0                                                          002010
      MSCPAD(1) = NRUN(5)                                               002020
      N = 2                                                             002030
      DO WHILE (N.LE.16)                                                002040
         MSCPAD(N) = 0                                                  002050
         N = N + 1                                                      002060
      END DO                                                            002070
      N = 1                                                             002080
      DO WHILE (N.LE.5)                                                 002090
         MICAD(N) = 0                                                   002100
         N = N + 1                                                      002110
      END DO                                                            002120
      MICOA(1) = 0                                                      002130
      MICOA(2) = 0                                                      002140
      MICOA(3) = 0                                                      002150
      FINIS = .FALSE.                                                   002160
      ERROR = .FALSE.                                                   002170
      MCNT = 0                                                          002180
      STIME = 0.0                                                       002190
      ITIME = 0                                                         002200
      IETIM = 0                                                         002210
      RETURN                                                            002220
      END                                                               002230
```

0 ERRORS FOUND IN THIS ROUTINE. 53 CARDS READ. 51 CARDS OUTPUT.

```
      SUBROUTINE DCODC                                                  002240
C                                                                       002250
C***** ***********************************************************     002260
C                                                                       002270
C     DECODE CONTROL MEMORY INSTRUCTION POINTED TO BY H-REG.            002280
C     SEPARATES ALL FIELDS INTO INDIVIDUAL WORDS IN THE VECTOR          002290
C     MICRO(25).  H-REG IS RESET BY FINAL CONTENTS OF MICRO(5).         002300
C                                                                       002310
C***** ***********************************************************     002320
C                                                                       002330
      LOGICAL   FINIS  , ERROR                                          002340
      COMMON    CNTMEM(1024)    , ICN    , ICOUNT , IDIN   ,          002350
     1          IDOUT  , IDSCR  , ITELM  , MAINM(1024)     ,          002360
     2          MBUSS  , MEMC   , MEMM   , MICAUT(6), MICOA(3),       002370
     3          MICQ   , MSCPAD(16)      , MSR1   , MSR2   ,          002380
     4          NRUN(20), PINES , ERROR  , MINSTR , MOPND  ,          002390
     5          MICRO(25),MCNT  , STIME  , ITIME  , IETIM             002400
      COMMON    KAD    , KDA    , IOLD3                               002410
      DATA MSK1 /68/                                                    002420
      IPTR = MEMM                                                       002430
      IERR = 0                                                          002440
      CALL GETCM (IPTR, 0, 5, MOP, IFLAG)                               002450
C     OPERATION CODE BITS 2 AND 3 STORED INVERTED, SO FIX:              002460
      MICRO(1) = XOR (MSK1, MOP)                                        002470
      IERR = IFLAG                                                      002480
      CALL GETCM (IPTR, 5, 3, MOP, IFLAG)                               002490
      MICRO(2) = MOP                                                    002500
      IERR = OR (IERR, IFLAG)                                           002510
      CALL GETCM (IPTR, 8, 6, MOP, IFLAG)                               002520
      MICRO(3) = MOP                                                    002530
      IERR = OR(IERR, IFLAG)                                            002540
      CALL GETCM (IPTR, 12, 4, MOP, IFLAG)                              002550
      MICRO(4) = MOP                                                    002560
      IERR = OR(IERR, IFLAG)                                            002570
      CALL GETCM (IPTR, 16, 10, MOP, IFLAG)                             002580
      MICRO(5) = MOP                                                    002590
      IERR = OR(IERR, IFLAG)                                            002600
      CALL GETCM (IPTR, 49, 3, MOP, IFLAG)                              002610
      MICRO(25) = MOP                                                   002620
      IERR = OR(IERR, IFLAG)                                            002630
      IFLAG = IERR                                                      002640
      NCOL = 26                                                         002650
      INDX = 6                                                          002660
      DO WHILE (IFLAG.EQ.0.AND.INDX.LE.24)                              002670
         CALL GETCM (IPTR, NCOL, 1, MOP, IFLAG)                         002680
         MICRO(INDX) = MOP                                              002690
         NCOL = NCOL + 1                                                002700
         INDX = INDX + 1                                                002710
      END DO                                                            002720
      IF (IFLAG.EQ.0)                                                   002730
         THEN                                                           002740
            IF (NRUN(3).EQ.2)                                           002750
               THEN                                                     002760
                  WRITE (7, 5) MEMM, MICRO(5)                           002770
    5             FORMAT (     15H1THIS INSTR AT: , O4, 12H, H-FIELD = , O4, 2H , ) 002780
            END IF                                                      002790
         ELSE                                                           002800
            WRITE (7, 10) IFLAG                                         002810
   10       FORMAT (43HCINVALID CALL TO GETCM FROM DCODC, IFLAG =  ,I2,1H, )  002820
            ERROR = .TRUE.                                              002830
      END IF                                                            002840
      RETURN                                                            002850
      END                                                               002860
```

0 ERRORS FOUND IN THIS ROUTINE. 53 CARDS READ. 59 CARDS OUTPUT.

```
      SUBROUTINE INMOD (IA, IB)                                          002870
                                                                         002880
C*****************************************************************     002890
C                                                                        002900
C     SIMULATE ACTIVITY GENERATED BY  INSTRUCTION MODIFIER, MICRO(2)     002910
C                                                                        002920
C*****************************************************************     002930
C                                                                        002940
      LOGICAL   FINIS  , ERROR                                           002950
      COMMON    CNTMEM(1024)     , ICN    , ICOUNT , IDIN   ,           002960
     1          IDOUT  , IDSCR   , ITELM  , MAINM(1024)     ,           002970
     2          MBUSS  , MEMC    , MEMM   , MICAO(6), MICDA(3),          002980
     3          MICQ   , MSCPAD(16)       , MSR1   , MSR2   ,           002990
     4          NRUN(20), FINIS  , ERROR  , NINSTR , NOPND  ,           003000
     5          MICRO(25),MCNT   , STIME  , ITIME  , IETIM              003010
      COMMON    KAD    , KDA     , IOLDB                                 003020
      DATA KMASK /1777778/                                               003030
      DATA IMIBIT/1000000/                                               003040
      IOP = MICRO(1)                                                     003050
      MOD = MICRO(2)                                                     003060
      IF (MOD.EQ.0)                                                      003070
         THEN                                                            003080
            IDOUT = MSCPAD(IB)                                           003090
      OR IF (MOD.EQ.1)                                                   003100
         THEN                                                            003110
            IDOUT = MSCPAD(IA)                                           003120
      OR IF (MOD.EQ.2)                                                   003130
         THEN                                                            003140
            IDOUT = IOLDB                                                003150
      OR IF (MOD.EQ.3)                                                   003160
         THEN                                                            003170
C           LEFT SHIFT OFF BIT FROM B GOES TO SR1:                       003180
            IBIT = SHIFT (MSCPAD(IB), -15)                               003190
            IF (IBIT.EQ.0)                                               003200
               THEN                                                      003210
                  MSR1 = 1                                               003220
               ELSE                                                      003230
                  MSR1 = 0                                               003240
            END IF                                                       003250
            MSCPAD(IB) = AND ((SHIFT(MSCPAD(IB),1)), KMASK)              003260
            IDOUT = MSCPAD(IB)                                           003270
      OR IF (MOD.EQ.4)                                                   003280
         THEN                                                            003290
C           RIGHT SHIFT OFF BIT FROM B                                   003300
            MSCPAD(IB) = SHIFT(MSCPAD(IB), -1)                           003310
            MSCPAD(IB) = OR(MSCPAD(IB), IMIBIT)                          003320
            IDOUT = MSCPAD(IB)                                           003330
      OR IF (MOD.EQ.5)                                                   003340
         THEN                                                            003350
C           LEFT SHIFT OFF BIT FROM Q-REG GOES TO LSB OF B:              003360
C           LEFT SHIFT OFF BIT FROM B GOES TO SR1:                       003370
            IBIT = SHIFT (MSCPAD(IB), -15)                               003380
            IF (IBIT.EQ.0)                                               003390
               THEN                                                      003400
                  MSR1 = 1                                               003410
               ELSE                                                      003420
```

```
                MSR1 = 0                                                    003430
            END IF                                                          003440
            IBIT = SHIFT (MICQ, -15)                                        003450
            MICQ = AND ( SHIFT(MICQ, 1) , KMASK )                           003460
            MICQ = OR(MICQ, 1)                                              003470
            MSCPAD(I8) = AND (SHIFT(MSCPAD(I8), 1) , KMASK)                 003480
            MSCPAD(I8) = OR (MSCPAD(I8), IBIT)                              003490
            IDOUT = MSCPAD(I8)                                              003500
      OR IF (MOD.EQ.6)                                                  003510
         THEN                                                             003520
C           RIGHT SHIFT OFF BIT FROM Q-REG TO SR1, LSB OF B INTO MSB OF Q:  003530
C           (IF,LOW-BIT FROM Q = 1, SR1 = 0; ELSE SR1 = 1)                  003540
            IBIT = AND (MSCPAD(I8), 1)                                      003550
            IBIT = SHIFT (IBIT, 15)                                         003560
            LOBIT = AND (MICQ, 1)                                           003570
            IF (LOBIT.EQ.0)                                                 003580
               THEN                                                         003590
                  MSR1 = 1                                                  003600
               ELSE                                                         003610
                  MSR1 = 0                                                  003620
            END IF                                                          003630
            MICQ = SHIFT (MICQ, -1)                                         003640
            MICQ = OR(MICQ, IBIT)                                           003650
            MSCPAD(I8) = SHIFT (MSCPAD(I3), -1)                             003660
            MSCPAD(I8) = OR(MSCPAD(I8), IHIBIT)                             003670
            IDOUT = MSCPAD(I8)                                              003680
         ELSE                                                           003690
            WRITE (7, 5) MOD                                                003700
    5       FORMAT (26H INVALID INSTR MODIFIER =   ,O4,15H.  RUN HALTED.  )  003710
            ERROR = .TRUE.                                                  003720
      END IF                                                            003730
      IF (NRUN(3).EQ.2) WRITE (7, 10) MOD, IDOUT                        003740
   10 FORMAT (13HOINSTR MOD =  ,I1,21H JUST SET DATA-OUT =  ,O8, 1H.)   003750
      IF (NRUN(3).EQ.2.OR.NRUN(3).EQ.3)                                 003760
         THEN                                                           003770
            CALL REPT2                                                  003780
      END IF                                                            003790
      RETURN                                                            003800
      END                                                               003810
```

0 ERRORS FOUND IN THIS ROUTINE. 95 CARDS READ. 103 CARDS OUTPUT.

```
      SUBROUTINE EXECM                                                  003820
C                                                                       003830
C***** ***********************************************************     003840
C                                                                      *003850
C     EXECUTE CONTROL MEMORY INSTRUCTION                               *003860
C                                                                      *003870
C***** ***********************************************************     003880
C                                                                       003890
      LOGICAL   FINIS  , ERROR                                          003900
      COMMON    CNTMEM(1024)     , ICN    , ICOUNT , IDIN   ,          003910
     1          IDOUT  , IDSCR   , ITELM  , MAINM(1024)     ,          003920
     2          MBUSS  , MEMC    , MEMT   , MICAD(6), MICDA(3),         003930
     3          MICQ   , MSCPAD(16)       , MSR1   , MSR2   ,          003940
     4          NRUN(20), FINIS  , ERROR  , NINSTR , NOPND  ,          003950
     5          MICRO(25),MCNT   , STIME  , ITIME  , IETIM             003960
      COMMON    KAD    , KDA    , IOLD                                  003970
C     EXECUTE APPLICABLE CONTROL FUNCTIONS BEFORE INSTR EXECUTION:      003980
      CALL PRBIN                                                        003990
      IF (MSR2.EQ.1)                                                    004000
        THEN                                                              004010
          MICRO(1) = OR (MICRO(1), 6)                                       004020
          IF (NRUN(3).EQ.2) WRITE (7, 23) MICRO(1)                          004030
   23     FORMAT (15HOOP CODE NOW =  , 02, 15H DUE TO SR2 = 1.   )          004040
      END IF                                                            004050
      IOP = MICRO(1)                                                    004060
      IF (IOP.GE.0.AND.IOP.LE.6)                                        004070
        THEN                                                              004080
          CALL BOLN (IOP)                                                   004090
      OR IF (IOP.GE.7.AND.IOP.LE.10)                                    004100
        THEN                                                              004110
          CALL INVRT (IOP)                                                  004120
      OR IF (IOP.GE.11.AND.IOP.LE.14)                                   004130
        THEN                                                              004140
          CALL CMPLT (IOP)                                                  004150
      OR IF (IOP.GE.15.AND.IOP.LE.18)                                   004160
        THEN                                                              004170
          CALL TRINC (IOP)                                                  004180
      OR IF (IOP.GE.19.AND.IOP.LE.22)                                   004190
        THEN                                                              004200
          CALL DECTR (IOP)                                                  004210
      OR IF (IOP.GE.23.AND.IOP.LE.26)                                   004220
        THEN                                                              004230
          CALL SUM   (IOP)                                                  004240
      OR IF (IOP.GE.27.AND.IOP.LE.31)                                   004250
        THEN                                                              004260
          CALL DIFF (IOP)                                                   004270
        ELSE                                                              004280
          WRITE (7, 10) IOP                                                 004290
   10     FORMAT (18HOINVALID OP CODE: , I2, 22H ENCOUNTERED IN EXECM.  )   004300
          ERROR = .TRUE.                                                    004310
      END IF                                                            004320
      CALL BINRY                                                        004330
      ICOUNT = ICOUNT + 1                                               004340
      RETURN                                                            004350
      END                                                               004360
```

0 ERRORS FOUND IN THIS ROUTINE. 55 CARDS READ. 62 CARDS OUTPUT.

```
      SUBROUTINE FINAL                                                  004370
C                                                                       004380
C***** ****************************************************************004390
C                                                                      *004400
C     EVALUATE RUN TERMINATION OPTIONS - SET FINIS = .TRUE. WHEN TRUE  *004410
C                                                                      *004420
C***** ****************************************************************004430
C                                                                       004440
      LOGICAL   FINIS  , ERROR                                          004450
      COMMON    CNTMEM(1024)     , ICN    , ICOUNT , IDIN   ,          004460
     1          IDOUT  , IDSCR  , ITELM  , MAINM(1024)     ,          004470
     2          MBUSS  , MEMC   , MEMM   , MICAD(6), MICDA(3),        004480
     3          MICQ   , MSCPAD(16)      , MSR1   , MSR2   ,          004490
     4          NRUN(20), FINIS  , ERR3C  , MINSTR , MOPND  ,         004500
     5          MICRO(25),MCNT   , STIME  , ITIME  , IETIM            004510
      COMMON    KAD    , KDA    , IOL)3                               004520
      NEND = NRUN(9)                                                    004530
      NVAL = NRUN(10)                                                   004540
      IF (NEND.EQ.1)                                                    004550
         THEN                                                           004560
            IF (MEMM.EQ.NVAL)                                           004570
               THEN                                                     004580
                  FINIS = .TRUE.                                        004590
            END IF                                                      004600
      OR IF (NEND.EQ.2)                                                 004610
         THEN                                                           004620
            IF (MEMC.EQ.NVAL)                                           004630
               THEN                                                     004640
                  FINIS = .TRUE.                                        004650
            END IF                                                      004660
      OR IF (NEND.EQ.3)                                                 004670
         THEN                                                           004680
            IF (ITIME.GE.NVAL)                                          004690
               THEN                                                     004700
                  FINIS = .TRUE.                                        004710
            END IF                                                      004720
      OR IF (NEND.EQ.4)                                                 004730
         THEN                                                           004740
            IF (IETIM.EQ.NVAL)                                          004750
               THEN                                                     004760
                  FINIS = .TRUE.                                        004770
            END IF                                                      004780
      OR IF (NEND.EQ.5)                                                 004790
         THEN                                                           004800
            IF (MCNT.GE.NVAL)                                           004810
               THEN                                                     004820
                  FINIS = .TRUE.                                        004830
            END IF                                                      004840
      OR IF (NEND.EQ.6)                                                 004850
         THEN                                                           004860
         ELSE                                                           004870
            WRITE (7, 5) NEND                                           004880
    5       FORMAT (31HCINVALID RUN TERMINATION CODE = , I3, 1H. )      004890
            ERROR = .TRUE.                                              004900
      END IF                                                            004910
      RETURN                                                            004920
      END                                                               004930
```

0 ERRORS FOUND IN THIS ROUTINE. 57 CARDS READ. 67 CARDS OUTPUT.

```
      SUBROUTINE SETMM                                                  004940
C                                                                       004950
C***** ***************************************************************004960
C                                                                      *004970
C     THIS ROUTINE PRESETS MAIN MEMORY TO A VALUE DESIGNATED BY THE    *004980
C     RUN CONTROL CARD, INCLUDING THE ABILITY TO SET EVERY LOCATION    *004990
C     TO BE A NO-OP POINTING TO ITSELF                                 *005000
C                                                                      *005010
C***** ***************************************************************005020
C                                                                       005030
      LOGICAL   FINIS  ,  ERROR                                         005040
      COMMON    CNTMEM(1024)     ,  ICN    ,  ICOUNT ,  IDIN   ,      005050
     1          IDOUT  ,  IDSCR  ,  ITELM  ,  MAINM(1024)      ,      005060
     2          MBUSS  ,  MEMC   ,  MEMI   ,  MICAD(6), MICDA(3),     005070
     3          MICQ   ,  MSCPAD(16)       ,  MSR1   ,  MSR2   ,      005080
     4          NRUN(20), FINIS  ,  ERROR  ,  MINSTR ,  MOPND  ,      005090
     5          MICRO(25),MCNT   ,  STIME  ,  ITIME  ,  IETIM         005100
      COMMON    KAD    ,  KDA    ,  IOLDB                             005110
      DATA MSK /3778/                                                   005120
      INDX = 1                                                          005130
      DO WHILE (.NOT.ERROR.AND.INDX.LE.1024)                            005140
         IF (NRUN(2).EQ.1)                                                 005150
            THEN                                                              005160
               MAINM(INDX) = 0                                                   005170
         OR IF (NRUN(2).EQ.2)                                              005180
            THEN                                                              005190
               NEWDX = INDX - 1                                                  005200
               MAINM(INDX) = AND (NEWDX, MSK)                                    005210
            ELSE                                                              005220
               WRITE (7, 10) NRUN(2)                                             005230
   10          FORMAT (44H0SETMM CALLED WITH INVALID OPTION NRUN(2) =  , I3, 1H.)005240
               ERROR = .TRUE.                                                    005250
         END IF                                                           005260
         INDX = INDX + 1                                                  005270
      END DO                                                            005280
      RETURN                                                            005290
      END                                                               005300
```

0 ERRORS FOUND IN THIS ROUTINE. 37 CARDS READ. 35 CARDS OUTPUT.

```
      SUBROUTINE SETCM                                                  005310
C                                                                       005320
C***** ************************************************************** *005330
C                                                                       *005340
C     THIS ROUTINE PRESETS CONTROL MEMORY TO A VALUE DESIGNATED BY THE  *005350
C     RUN CONTROL CARD, INCLUDING THE ABILITY TO SET EVERY LOCATION     *005360
C     TO BE A NO-OP POINTING TO ITSELF.                                 *005370
C     REPL A DOUBLE-WORD ZERO, REPL2 IS ALIGNED TO INCREMENT THE        *005380
C     H-FIELD BY 1, PRESET -1 SO 1ST WILL BE 0, AND LOAD NEXT H BIT ON  *005390
C                                                                       *005400
C***** ************************************************************** 005410
C                                                                       005420
      LOGICAL   FINIS  ,  ERROR                                         005430
      COMMON    CNTMEM(1024)     ,  ICN    ,  ICOUNT ,  IDIN   ,      005440
     1          IDOUT  ,  IDSCR  ,  ITELM  ,  MAINM(1024)      ,      005450
     2          MBUSS  ,  MEMC   ,  MEMH   ,  MICAD(6), MICDA(3),     005460
     3          MICQ   ,  MSCPAD(16)       ,  MSR1   ,  MSR2   ,      005470
     4          NRUN(20), FINIS  ,  ERROR  ,  MINSTR ,  MOPND  ,      005480
     5          MICRO(25),MCNT   ,  STIMEI ,  ITIME  ,  IETIM         005490
      COMMON    KAD    ,  KDA    ,  IOLDB                             005500
      DATA REPL /0B/, REPL2 /53777777620.0000B/                         005510
      DATA ADDR /2000C0008/                                             005520
      INDX = 1                                                          005530
      DO WHILE (.NOT.ERROR.AND.INDX.LE.1024)                            005540
         IF (NRUN(1).EQ.1)                                              005550
            THEN                                                        005560
               CNTMEM(INDX) = REPL                                      005570
         OR IF (NRUN(1).EQ.2)                                           005580
            THEN                                                        005590
               REPL2 = REPL2 + ADDR                                     005600
               CNTMEM(INDX) = REPL2                                     005610
            ELSE                                                        005620
               WRITE (7, 10) NRUN(1)                                    005630
   10          FORMAT (44HCSETCM CALLED WITH INVALID OPTION NRUN(1) =  , I3,1H. )005640
               ERROR = .TRUE.                                           005650
         END IF                                                         005660
         INDX = INDX + 1                                                005670
      END DO                                                            005680
      RETURN                                                            005690
      END                                                               005700

    0 ERRORS FOUND IN THIS ROUTINE.    40 CARDS READ.          36 CARDS OUTPUT.
```

```
      SUBROUTINE LOADM                                                  005710
C                                                                       005720
C***** ****************************************************************005730
C                                                                      *005740
C     CARD LAYOUT: COL  1 = 1, MAIN MEMORY                             *005750
C                  COL  1 = 2, CONTROL MEMORY                          *005760
C                  COL  1 = 9, LOAD COMPLETED                          *005770
C     CONTIGUOUS MEMORY IMAGES (NRUN(11) = 1):                         *005780
C               COLS 5-12 = ADDRESS, OCTAL                             *005790
C     COLS 13-20 = MAIN MEMORY CONTENTS - OCTAL                        *005800
C     COLS 21-36 = CONTROL MEMORY CONTENTS - OCTAL                     *005810
C     MICRO-FIELD IMAGES (NRUN(11) = 2):                               *005820
C     COLS 3-6 = ADDRESS, OCTAL                                        *005830
C     COLS 15-16 = INSTRUCTION (OP CODE - OCTAL)                       *005840
C     COL  17 = INSTRUCTION MODIFIER (OCTAL)                           *005850
C     COLS 18-19 = A OPERAND (OCTAL)                                   *005860
C     COLS 20-21 = B OPERAND (OCTAL)                                   *005870
C     COLS 22-25 = NEXT M ADDRESS (OCTAL)                              *005880
C     COLS 26-33 = BINARY CONTROL FIELDS PLUS A/D MUX ADDR             *005890
C                                                                      *005900
C***** ****************************************************************005910
C                                                                       005920
      LOGICAL   FINIS  , ERROR                                          005930
      COMMON    CNTMEM(1024)     , IGN    , ICOUNT , IDIN   ,          005940
     1          IDOUT  , IDSCR   , ITELM  , MAINM(1024)     ,          005950
     2          MBUSS  , MEMC    , MEM+   , MICAD(6), MICDA(3),        005960
     3          MICQ   , MSCPAD(16)       , MSR1   , MSR2   ,          005970
     4          NRUN(20), FINIS  , ERROR  , MINSTR , MOPNO  ,          005980
     5          MICRO(25),MCNT   , STIME  , ITIME  , IETIM             005990
      COMMON    KAD    , KDA    , IOLDB                                006000
      DIMENSION CNTFLD(6)                                               006010
      DATA MSK1 /1400000000000000038/                                   006020
      KEY = 0                                                           006030
      K = 0                                                             006040
      L = 0                                                             006050
      IF (NRUN(11).EQ.1)                                                006060
         THEN                                                           006070
            DO WHILE (KEY.NE.9)                                         006080
               READ (5, 10) KEY, MADR, MAIN, CNTRL                      006090
   10          FORMAT (I1, 7X, O4, 2X, O6, O16)                         006100
               IF (KEY.EQ.1)                                            006110
                  THEN                                                  006120
                     MAINM(MADR+1) = MAIN                               006130
                     K = K + 1                                          006140
               OR IF (KEY.EQ.2)                                         006150
                  THEN                                                  006160
C                    INVERT BITS 2,3 OF OP CODE (0-+):                  006170
                     CNTMEM(MADR+1) = XOR (MSK1, CNTRL)                 006180
                     L = L + 1                                          006190
               OR IF (KEY.EQ.9)                                         006200
                  THEN                                                  006210
                     WRITE (7, 23) K,L                                  006220
   20                FORMAT (24HOMEMORY LOAD COMPLETED : , I5, 19H WORDS OF MAIN AND  ,006230
     .                        I5, 32H WORDS OF CONTROL MEMORY LOADED.  ,//)006240
                  ELSE                                                  006250
                     WRITE (7, 33) KEY                                  006260
```

```
30                FORMAT (49HJINVALID CONTROL CHARACTER IN MEMORY LOAD CARD =    ,    006270
     .                        I1, 18H.  RUN TERMINATED.  )                          006280
                  KEY = 9                                                          006290
                  ERROR = .TRUE.                                                   006300
               END IF                                                           006310
            END DO                                                           006320
         OR IF (NRUN(11).EQ.2)                                            006330
         THEN                                                             006340
            DO WHILE (KEY.NE.9)                                              006350
               READ (5, 40) KEY, MADR, MAINL, MAINR, (CNTFLD(I), I = 1,6)       006360
+0             FORMAT (I1,1X,04,1X,03,03,1X,02,01,02,02,04,08)                  006370
               IF (KEY.EQ.1)                                                    006380
                  THEN                                                             006390
                     MAINM(MADR+1) = OR (SHIFT (MAINL, 8), MAINR)                  006400
                     K = K + 1                                                     006410
               OR IF (KEY.EQ.2)                                                 006420
                  THEN                                                             006430
                     CNTRL = OR (SHIFT (CNTFLD(1), 3), CNTFLD(2) )                 006440
                     CNTRL = OR (SHIFT (CNTRL, 4), CNTFLD(3) )                     006450
                     CNTRL = OR (SHIFT (CNTRL, 4), CNTFLD(4) )                     006460
                     CNTRL = OR (SHIFT (CNTRL, 10), CNTFLD(5) )                    006470
                     CNTRL = OR (SHIFT (CNTRL, 22), CNTFLD(6) )                    006480
C                    INVERT BITS 2,3 OF OP CODE (0,4):                             006490
                     CNTMEM(MADR+1) = XOR (MSK1, CNTRL)                            006500
                     L = L + 1                                                     006510
               OR IF (KEY.EQ.9)                                                 006520
                  THEN                                                             006530
                     WRITE (7, 20) K, L                                            006540
                  ELSE                                                             006550
                     WRITE (7, 30) KEY                                             006560
                     KEY = 9                                                       006570
                     ERROR = .TRUE.                                                006580
               END IF                                                           006590
            END DO                                                           006600
         ELSE                                                             006610
            WRITE (7, 50) NRUN(11)                                           006620
50          FORMAT (20HOINVALID NRUN(11) =    ,04, 18H.  RUN TERMINATED.    )  006630
            ERROR = .TRUE.                                                   006640
      END IF                                                              006650
      RETURN                                                              006660
      END                                                                 006670
```

0 ERRORS FOUND IN THIS ROUTINE. 971 CARDS READ. 96 CARDS OUTPUT.

```
      SUBROUTINE GETCM (IADR, IBITL, IBITN, IBYTE, IFLAG)                  006680
C                                                                          006690
C***** ***************************************************************** *006700
C                                                                          *006710
C     USED TO EXTRACT A FIELD FROM THE SIMULATED 48-BIT MICRO-             *006720
C     PROCESSOR CONTROL MEMORY - PARAMETERS -                              *006730
C        IADR - RELATIVE CONTROL MEMORY LOCATION - 0 TO 1023               *006740
C        IBITL - STARTING BIT OF DESIRED FIELD - 0 TO 47                   *006750
C        IBITN - NUMBER OF BITS IN FIELD - 1 TO 48  (24 MAX RETURNED)      *006760
C        IBYTE - RETURN WORD, FIELD RIGHT ADJUSTED, ZERO FILLED            *006770
C        IFLAG - ERROR RETURNS - ZERO, NO ERROR - 1,2, OR 3 -ERROR IN      *006780
C            PARAMETER 1,2, OR 3.  CODE 3, BITS WRAPAROUND WORD            *006790
C                                                                          *006800
C***** ***************************************************************** *006810
C                                                                          006820
      LOGICAL   FINIS  , ERROR                                             006830
      COMMON    CNTMEM(1024)     , ICN    , ICOUNT , IOIN   ,             006840
     1          IOOUT  , IOSCR   , ITELY  , %AINM(1024)     ,             006850
     2          MBUSS  , MEMC    , MEMM   , %ICAD(6), MICOA(3),           006860
     3          MICQ   , MSCPAD(16)       , MSR1   , MSR2   ,             006870
     4          NRUN(20), FINIS  , ERROR  , %INSTR , MOPND  ,             006880
     5          MICRO(25),MCNT   , STIME  , ETIME  , IETIM                006890
      COMMON    KAD    , KDA     , IOLDB                                   006900
C     TEST PARAMETERS FOR VALIDITY                                         006910
      IF (IADR.LT.0.OR.IADR.GT.1023)                                       006920
         THEN                                                                006930
            IFLAG = 1                                                        006940
      OR IF (IBITL.LT.0.OR.IBITL.GT.47)                                    006950
         THEN                                                                006960
            IFLAG = 2                                                        006970
      OR IF (IBITN.LT.0.OR.IBITN.GT.48)                                    006980
         THEN                                                                006990
            IFLAG = 3                                                        007000
      OR IF((IBITL + IBITN).GT.49)                                         007010
         THEN                                                                007020
            IFLAG = 9                                                        007030
         ELSE                                                                007040
            IFLAG = 0                                                        007050
C           GET FIELD -                                                      007060
C           ADJUST START BIT FOR 60-BIT WORD                                 007070
            IDR = IADR + 1                                                   007080
            ISTBIT = 48 - IBITL                                              007090
            IMSK = SHIFT (MASK(IBITN), ISTBIT)                               007100
            IBYTE = AND (CNTMEM(IDR), IMSK)                                  007110
C           RIGHT ADJUST                                                     007120
            NMBR = 48 -(IBITL + IBITN)                                       007130
            IBYTE = SHIFT (IBYTE, -NMBR)                                     007140
      END IF                                                               007150
      IF (IFLAG.NE.0)                                                      007160
         THEN                                                                007170
            IBYTE = 0                                                        007180
      END IF                                                               007190
      RETURN                                                               007200
      END                                                                  007210

0 ERRORS FOUND IN THIS ROUTINE.    3. CARDS READ.          45 CARDS OUTPUT.
```

```
      SUBROUTINE DCODM                                                  007220
C                                                                       007230
C***** ***********************************************************     007240
C                                                                      *007250
C     DECODE MAIN MEMORY INSTRUCTION POINTED TO BY PC (MSCPAD(1)       *007260
C     RESET PC = PC + 1.  LOADH-REG FROM 1ST HALF, C-REG               *007270
C     LOADED FROM 2ND HALF OF MAIN MEMORY WORD.                        *007280
C                                                                      *007290
C***** ***********************************************************     007300
C                                                                       007310
      LOGICAL   FINIS  , ERROR                                          007320
      COMMON    CNTMEM(1024)     , ICN    , ICOUNT , IDIN   ,          007330
     1          IDOUT  , IDSCR , ITELM , MAINM(1024)     ,           007340
     2          MBUSS  , MEMC  , MEMM  , MICAD(6), MICDA(3),          007350
     3          MICQ   , MSCPAD(16)      , MSR1   , MSR2   ,          007360
     4          NRUN(20), FINIS , ERROR , MINSTR , MOPND  ,          007370
     5          MICRO(25),MCNT  , STIME , ITIME  , IETIM             007380
      COMMON    KAD    , KDA   , IOLD                                  007390
      DATA IOFF /1400B/                                                 007400
      MEMC = MSCPAD(1)                                                  007410
      MSCPAD(1) = MSCPAD(1) + 1                                         007420
      CALL GETMM (MEMC, 0, 8, MIN, IFLAG)                               007430
      MINSTR = MIN                                                      007440
      IF (IFLAG.EQ.0)                                                   007450
         THEN                                                              007460
            MEMM = MINSTR + IOFF                                           007470
            CALL GETMM (MEMC, 8, 8, MOP, IFLAG)                            007480
            MOPND = MOP                                                    007490
            IF (IFLAG.EQ.0)                                                007500
               THEN                                                           007510
                  MEMC = MOPND + IOFF                                            007520
               ELSE                                                           007530
                  WRITE (7, 10) IFLAG                                            007540
   10             FORMAT (43H0INVALID CALL TO GETMM FROM DCODM, IFLAG = , I2, 1H. ) 007550
                  ERROR = .TRUE.                                                 007560
            END IF                                                         007570
         ELSE                                                              007580
            WRITE (7, 10) IFLAG                                            007590
            ERROR = .TRUE.                                                 007600
      END IF                                                            007610
      RETURN                                                            007620
      END                                                               007630

     0 ERRORS FOUND IN THIS ROUTINE.    42 CARDS READ.          37 CARDS OUTPUT.
```

```
      SUBROUTINE GETMM (IADR, IBITL, IBITN, IBYTE, IFLAG)                  007640
C                                                                          007650
C***** ********************************************************************007660
C                                                                          *007670
C     USED TO EXTRACT A FIELD FROM THE SIMULATED 16-BIT MICRO-             *007680
C     PROCESSOR MAIN MEMORY.  PARAMETERS:                                  *007690
C     IADR - RELATIVE MEMORY LOCATION - BASE ZERO                          *007700
C     IBITL - STARTING BIT OF DESIRED FIELD - 1 TO 15                      *007710
C     IBITN - NUMBER OF BITS IN FIELD                                      *007720
C     IBYTE - RETURN LOCATION - RIGHT ADJUSTED, ZERO FILLED                *007730
C     IFLAG - ERROR CONDITIONS - ZERO, NONE.  1-3, CORRESPONDS TO          *007740
C             INVALID PARAMETER 1-3   4= BITS WRAPAROUND                   *007750
C                                                                          *007760
C***** ********************************************************************007770
C                                                                          007780
      LOGICAL   FINIS  , ERROR                                             007790
      COMMON    CNTMEM(1024)     , ICN    , ICOUNT , IOIN   ,             007800
     1          IOOUT  , IOSCR   , ITELM  , MAINM(1024)     ,             007810
     2          MBUSS  , MEMC    , MEMM   , MICAD(6), MICDA(3),            007820
     3          MICQ   , MSCPAD(16)       , MSR1   , MSR2   ,             007830
     4          NRUN(20), FINIS  , ERROR  , MINSTR , MOPND  ,             007840
     5          MICRO(25),MCNT   , STIME  , ITIME  , IETIM                007850
      COMMON    KAD    , KDA     , IOLJB                                   007860
C     TEST PARAMETERS FOR VALIDITY                                         007870
      IF (IADR.LT.0.OR.IADR.GT.1023)                                       007880
         THEN                                                              007890
            IFLAG = 1                                                      007900
      OR IF (IBITL.LT.0.OR.IBITL.GT.15)                                    007910
         THEN                                                              007920
            IFLAG = 2                                                      007930
      OR IF (IBITN.LT.0.OR.IBITN.GT.16)                                    007940
         THEN                                                              007950
            IFLAG = 3                                                      007960
      OR IF ((IBITL + IBITN).GT.16)                                        007970
         THEN                                                              007980
            IFLAG = 5                                                      007990
         ELSE                                                              008000
            IFLAG = 0                                                      008010
            IOR = IADR + 1                                                 008020
            IBIT = 16 - IBITL                                              008030
            IMASK = SHIFT (MASK(IBITN), IBIT)                              008040
            IBYTE = AND (MAINM(IOR), IMASK)                                008050
            NMBR = 16 - (IBITL + IBITN)                                    008060
            IBYTE = SHIFT (IBYTE, -NMBR)                                   008070
      END IF                                                               008080
      IF (IFLAG.NE.0)                                                      008090
         THEN                                                              008100
            IBYTE = 0                                                      008110
      END IF                                                               008120
      RETURN                                                               008130
      END                                                                  008140

0 ERRORS FOUND IN THIS ROUTINE.    51 CARDS READ.          45 CARDS OUTPUT.
```

```
      SUBROUTINE BOLN (IOP)                                              008150
C                                                                        008160
C***** ***************************************************************008170
C                                                                       *008180
C     ALU SIMULATION FOR OP CODES ZERO THRU SIX (BOOLEAN)               *008190
C                                                                       *008200
C***** ***************************************************************008210
C                                                                        008220
      LOGICAL   FINIS  , ERROR                                           008230
      COMMON    CNTMEM(1024)     , ICN    , ICOUNT , IDIN   ,           008240
     1          IDOUT  , IDSCR  , ITELM  , MAINM(1024)     ,           008250
     2          MBUSS  , MEMC   , MEMM   , MICAD(6), MICOA(3),         008260
     3          MICQ   , MSCPAD(16)      , MSR1   , MSR2   ,           008270
     4          NRUN(20), FINIS  , ERROR  , NINSTR , NOPND  ,          008280
     5          MICRO(25),MCNT   , STIME  , ITIME  , IETIM             008290
      COMMON    KAD    , KDA    , IOLDB                                 008300
      DATA LHIGH /1777778/                                               008310
      IF (NRUN(3).EQ.2) WRITE (7,  5)                                    008320
    5 FORMAT (27H0SUBROUTINE BOLN  ENTERED.  )                           008330
      MOD = MICRO(2)                                                     008340
      IAR = MICRO(3) + 1                                                 008350
      IBR = MICRO(4) + 1                                                 008360
      IOLDB = MSCPAD(IBR)                                                008370
      IF (IOP.EQ.0)                                                      008380
         THEN                                                              008390
            IF (ICN.EQ.0)                                                008400
               THEN                                                        008410
                  ITMP = LHIGH                                           008420
            OR IF (ICN.EQ.1)                                             008430
               THEN                                                        008440
                  ITMP = 0                                               008450
               ELSE                                                      008460
                  ERROR = .TRUE.                                         008470
                  WRITE (7, 10) ICN, IOP                                 008480
   10             FORMAT (17H0INVALID CARRY = ,O8,20H IN BOLN, OP CODE = , I2, 1H.) 008490
            END IF                                                       008500
      OR IF (IOP.EQ.1)                                                   008510
         THEN                                                            008520
            ITMP = AND (MSCPAD(IAR), MSCPAD(IBR) )                       008530
      OR IF (IOP.EQ.2)                                                   008540
         THEN                                                            008550
            ITMP = AND (IDIN, MSCPAD(IBR) )                              008560
      OR IF (IOP.EQ.3)                                                   008570
         THEN                                                            008580
            ITMP = OR (MSCPAD(IAR), MSCPAD(IBR) )                        008590
      OR IF (IOP.EQ.4)                                                   008600
         THEN                                                            008610
            ITMP = OR (IDIN, MSCPAD(IBR) )                               008620
      OR IF (IOP.EQ.5)                                                   008630
         THEN                                                            008640
            ITMP = XOR (MSCPAD(IAR), MSCPAD(IBR) )                       008650
      OR IF (IOP.EQ.6)                                                   008660
         THEN                                                            008670
            ITMP = XOR (IDIN, MSCPAD(IBR) )                              008680
      END IF                                                             008690
      IF (NRUN(3).EQ.2)                                                  008700
```

```
         THEN                                                                  008710
            WRITE (7, 25) IOP, ITMP                                            008720
25          FORMAT ( 7H0INSTR , 02, 24H EXECUTED, TEMP VALUE =  ,06,  1H.)     008730
      END IF                                                                   008740
      IF (MOD.EQ.7)                                                            008750
         THEN                                                                  008760
            MICQ = ITMP                                                        008770
            IOOUT = ITMP                                                       008780
         ELSE                                                                  008790
            MSCPAD(IBR) = ITMP                                                 008800
            CALL INMOD (IAR, IBR)                                              008810
      END IF                                                                   008820
      RETURN                                                                   008830
      END                                                                      008840
```

0 ERRORS FOUND IN THIS ROUTINE. 7:: CARDS READ. 84 CARDS OUTPUT.

```
PREPROCESSOR FOR STRUCTURED FORTRAN IV PROGRAMS     U.S. ARMY MISSLE COMMAND, AMSMI-RCG(WISE)      STR 1.2

      SUBROUTINE INVRT (IOP)                                                  000850
C                                                                             000860
C***** ********************************************************************* 000870
C                                                                             000880
C     ALU SIMULATION FOR OP CODES 7 THRU 10 (INVERT)                          000890
C                                                                             000900
C***** ********************************************************************* 000910
C                                                                             000920
      LOGICAL   FINIS  , ERROR                                                000930
      COMMON    CNTMEM(1024)     , ICN    , ICOUNT , IOIN   ,                000940
     1          IOOUT  , IOSCR   , ITEL4  , MAINM(1024)     ,                000950
     2          MBUSS  , MEMC    , MEMM   , MICAD(6), MICDA(3),              000960
     3          MICQ   , MSCPAD(16)       , MSR1   , MSR2   ,                000970
     4          NRUN(20), FINIS  , ERROR  , MINSTR , MOPND  ,                000980
     5          MICRO(25),MCNT   , STIME  , ITIME  , IETIM                   000990
      COMMON    KAD    , KDA     , IOLDB                                      001000
      DATA LHIGH /1777778/                                                    001010
      IF (NRUN(3).EQ.2) WRITE (7,  9)                                         001020
    9 FORMAT (27HOSUBROUTINE INVRT ENTERED.  )                                001030
      MOD = MICRO(2)                                                          001040
      IAR = MICRO(3) + 1                                                      001050
      IBR = MICRO(4) + 1                                                      001060
      IOLDB = MSCPAD(IBR)                                                     001070
      IF (IOP.EQ.7)                                                           001080
         THEN                                                                 001090
            ITMP = COMPL(MSCPAD(IAR) )                                        001100
      OR IF (IOP.EQ.8)                                                        001110
         THEN                                                                 001120
            ITMP = COMPL (IOIN)                                               001130
      OR IF (IOP.EQ.9)                                                        001140
         THEN                                                                 001150
            ITMP = COMPL (MSCPAD(IBR) )                                       001160
      OR IF (IOP.EQ.10)                                                       001170
         THEN                                                                 001180
            ITMP = COMPL (MICQ)                                               001190
      END IF                                                                  001200
      ITMP = AND(ITMP, LHIGH)                                                 001210
      IF (ICN.EQ.0)                                                           001220
         THEN                                                                 001230
            ITMP = ITMP + LHIGH                                               001240
            ITMP = AND (ITMP, LHIGH)                                          001250
      OR IF (ICN.LT.0.OR.ICN.GT.1)                                            001260
         THEN                                                                 001270
            WRITE (7, 10) ICN, IOP                                            001280
   10       FORMAT (17HOINVALID CARRY = ,I8,21H IN INVRT, OP CODE = ,I2,1H.)  001290
            ERROR = .TRUE.                                                    001300
      END IF                                                                  001310
      IF (NRUN(3).EQ.2)                                                       001320
         THEN                                                                 001330
            WRITE (7, 25) IOP, ITMP                                           001340
   25       FORMAT ( 7HOINSTR , I2, 24H EXECUTED, TEMP VALUE =  ,O6,  1H.)    001350
      END IF                                                                  001360
      IF (MOD.EQ.7)                                                           001370
         THEN                                                                 001380
            MICQ = ITMP                                                       001390
            IOOUT = ITMP                                                      001400
         ELSE                                                                 001410
            MSCPAD(IBR) = ITMP                                                001420
            CALL INMOD (IAR, IBR)                                             001430
      END IF                                                                  001440
      RETURN                                                                  001450
      END                                                                     001460

     0 ERRORS FOUND IN THIS ROUTINE.    91 CARDS READ.          78 CARDS OUTPUT.
```

```
      SUBROUTINE CMPLT (IOP)                                                 009470
C                                                                            009480
C***** ********************************************************************009490
C                                                                           *009500
C     ALU SIMULATION FOR OP CODES 11 THRU 14 (TWO'S COMPLEMENT)             *009510
C                                                                           *009520
C***** ********************************************************************009530
C                                                                            009540
      LOGICAL   FINIS  , ERROR                                               009550
      COMMON    CNTMEM(1024)     , ICN    , ICOUNT , IDIN   ,               009560
     1          IDOUT  , IDSCR   , ITELM  , MAINM(1024)     ,               009570
     2          MBUSS  , MEMC    , MEMM   , MICAD(6), MICDA(3),             009580
     3          MICQ   , MSCPAD(16)       , MSR1   , MSR2   ,               009590
     4          NRUN(20), FINIS  , ERROR  , NINSTR , NOPND  ,               009600
     5          MICRO(25),MCNT   , STIMEI , ITIME  , IETIM                  009610
      COMMON    KAD    , KDA     , IOLD3                                     009620
      DATA LHIGH /1777778/                                                   009630
      IF (NRUN(3).EQ.2) WRITE (7,  5)                                        009640
    5 FORMAT (27H0SUBROUTINE COMPL ENTERED.  )                               009650
      MOD = MICRO(2)                                                         009660
      IAR = MICRO(3) + 1                                                     009670
      IBR = MICRO(4) + 1                                                     009680
      IOLD3 = MSCPAD(IBR)                                                    009690
      IF (IOP.EQ.11)                                                         009700
         THEN                                                                009710
            ITMP = COMPL(MSCPAD(IAR))                                        009720
      OR IF (IOP.EQ.12)                                                      009730
         THEN                                                                009740
            ITMP = COMPL (IDIN)                                              009750
      OR IF (IOP.EQ.13)                                                      009760
         THEN                                                                009770
            ITMP = COMPL (MSCPAD(IBR))                                       009780
      OR IF (IOP.EQ.14)                                                      009790
         THEN                                                                009800
            ITMP = COMPL (MICQ)                                              009810
      END IF                                                                 009820
      ITMP = AND (ITMP, LHIGH)                                               009830
      ITMP = ITMP + ICN                                                      009840
      ITMP = AND (ITMP, LHIGH)                                               009850
      IF (NRUN(3).EQ.2)                                                      009860
         THEN                                                                009870
            WRITE (7, 25) IOP, ITMP                                          009880
   25       FORMAT ( 7H0INSTR , 02, 24H EXECUTED, TEMP VALUE =  ,06,  1H.)   009890
      END IF                                                                 009900
      IF (MOD.EQ.7)                                                          009910
         THEN                                                                009920
            MICQ = ITMP                                                      009930
            IDOUT = ITMP                                                     009940
         ELSE                                                                009950
            MSCPAD(IBR) = ITMP                                               009960
            CALL INMOD (IAR, IBR)                                            009970
      END IF                                                                 009980
      RETURN                                                                 009990
      END                                                                    010000

     0 ERRORS FOUND IN THIS ROUTINE.    57 CARDS READ.          58 CARDS OUTPUT.
```

```
      SUBROUTINE TRINC (IOP)                                                    010010
C                                                                               010020
C***** ***************************************************************         010030
C                                                                               *010040
C     ALU SIMULATION FOR OP CODES 15 THRU 18 (TRANSFER/INCREMENT)               *010050
C                                                                               *010060
C***** ***************************************************************         010070
C                                                                               010080
      LOGICAL   FINIS  , ERROR                                                  010090
      COMMON    CNTMEM(1024)     ,  ICN    ,  ICOUNT ,  IDIN   ,              010100
     1          IDOUT  ,  IDSCR  ,  ITEL4  ,  MAINM(1024)     ,              010110
     2          MBUSS  ,  MEMC   ,  MEMM   ,  MICAD(6), MICDA(3),            010120
     3          MICQ   ,  MSCPAD(16)        ,  MSR1   ,  MSR2   ,            010130
     4          NRUN(20), PINIS  ,  ERROR  ,  NINSTR ,  NOPND  ,            010140
     5          MICRO(25),MCNT   ,  STIME  ,  ITIME  ,  IETIM               010150
      COMMON    KAD    ,  KDA    ,  IOLDB                                     010160
      DATA LHIGH /1777778/                                                      010170
      IF (NRUN(3).EQ.2) WRITE (7,  5)                                           010180
    5 FORMAT (27H0SUBROUTINE TRINC ENTERED.  )                                  010190
      MOD = MICRO(2)                                                            010200
      IAR = MICRO(3) + 1                                                        010210
      IBR = MICRO(4) + 1                                                        010220
      IOLDB = MSCPAD(IBR)                                                       010230
      IF (IOP.EQ.15)                                                            010240
         THEN                                                                   010250
            ITMP = MSCPAD(IAR) + ICN                                            010260
      OR IF (IOP.EQ.16)                                                         010270
         THEN                                                                   010280
            ITMP = IDIN + ICN                                                   010290
      OR IF (IOP.EQ.17)                                                         010300
         THEN                                                                   010310
            ITMP = MSCPAD(IBR) + ICN                                            010320
      OR IF (IOP.EQ.18)                                                         010330
         THEN                                                                   010340
            ITMP = MICQ + ICN                                                   010350
      END IF                                                                    010360
      ITMP = AND (ITMP, LHIGH)                                                  010370
      IF (NRUN(3).EQ.2)                                                         010380
         THEN                                                                   010390
            WRITE (7, 25) IOP, ITMP                                             010400
   25       FORMAT (7H0INSTR  ,02,24H EXECUTED, TEMP VALUE =  ,08, 1H.)         010410
      END IF                                                                    010420
      IF (MOD.EQ.7)                                                             010430
         THEN                                                                   010440
            MICQ = ITMP                                                         010450
            IDOUT = ITMP                                                        010460
         ELSE                                                                   010470
            MSCPAD(IBR) = ITMP                                                  010480
            CALL INMOD (IAR, IBR)                                               010490
      END IF                                                                    010500
      RETURN                                                                    010510
      END                                                                       010520

    0 ERRORS FOUND IN THIS ROUTINE.    32 CARDS READ.          56 CARDS OUTPUT.
```

```
      SUBROUTINE DECTR (IOP)                                                    010530
C                                                                               010540
C***** ********************************************************************** 010550
C                                                                              *010560
C     ALU SIMULATION FOR OP CODES 19 THRU 22 (DECREMENT/TRANSFER)              *010570
C                                                                              *010580
C***** ********************************************************************** 010590
C                                                                               010600
      LOGICAL   FINIS  , ERROR                                                  010610
      COMMON    CNTMEM(1024)     , ICN    , ICOUNT , IDIN   ,                  010620
     1          IDOUT  , IDSCR   , ITELM  , MAINM(1024)     ,                  010630
     2          MBUSS  , MEMC    , MEMM   , MICAD(6), MICDA(3),               010640
     3          MICQ   , MSCPAD(16)       , MSR1   , MSR2   ,                  010650
     4          NRUN(20), FINIS  , ERROR  , MINSTR , MOPND  ,                  010660
     5          MICRO(25),MCNT   , STIME  , ITIME  , IETIM                     010670
      COMMON    KAD    , KDA     , IOLDB                                        010680
      DATA LHIGH /177777B/                                                      010690
      IF (NRUN(3).EQ.2) WRITE (7, 5)                                            010700
    5 FORMAT (27H0SUBROUTINE DECTR ENTERED.  )                                  010710
      MOD = MICRO(2)                                                            010720
      IAR = MICRO(3) + 1                                                        010730
      IBR = MICRO(4) + 1                                                        010740
      IOLDB = MSCPAD(IBR)                                                       010750
      IF (IOP.EQ.19)                                                            010760
         THEN                                                                    010770
            ITMP = MSCPAD(IAR)                                                   010780
      OR IF (IOP.EQ.20)                                                         010790
         THEN                                                                    010800
            ITMP = IDIN                                                          010810
      OR IF (IOP.EQ.21)                                                         010820
         THEN                                                                    010830
            ITMP = MSCPAD(IBR)                                                   010840
      OR IF (IOP.EQ.22)                                                         010850
         THEN                                                                    010860
            ITMP = MICQ                                                          010870
      END IF                                                                    010880
      IF (ICN.EQ.8)                                                             010890
         THEN                                                                    010900
            ITMP = ITMP + LHIGH                                                  010910
            ITMP = AND (ITMP, LHIGH)                                             010920
      OR IF (ICN.LT.0.OR.ICN.GT.1)                                              010930
         THEN                                                                    010940
            WRITE (7, 10) ICN, IOP                                               010950
   10       FORMAT (17H0INVALID CARRY = ,O8,21H IN DECTR, OP CODE = ,I2,1H.)     010960
      END IF                                                                    010970
      IF (NRUN(3).EQ.2)                                                         010980
         THEN                                                                    010990
            WRITE (7, 25) IOP, ITMP                                              011000
   25       FORMAT ( 7H0INSTR , I2, 24H EXECUTED, TEMP VALUE =  ,O6,  1H.)       011010
      END IF                                                                    011020
      IF (MOD.EQ.7)                                                             011030
         THEN                                                                    011040
            MICQ = ITMP                                                          011050
            IDOUT = ITMP                                                         011060
         ELSE                                                                    011070
            MSCPAD(IBR) = ITMP                                                   011080
            CALL INMOD (IAR, IBR)                                                011090
      END IF                                                                    011100
      RETURN                                                                    011110
      END                                                                       011120
```

0 ERRORS FOUND IN THIS ROUTINE. 51 CARDS READ. 68 CARDS OUTPUT.

```
      SUBROUTINE SUM (IOP)                                                  011130
C                                                                           011140
C***** ***************************************************************** 011150
C                                                                          *011160
C     ALU SIMULATION FOR OP CODES 23 THRU 26 (ADD)                         *011170
C                                                                          *011180
C***** ***************************************************************** 011190
C                                                                           011200
      LOGICAL   FINIS  , ERROR                                              011210
      COMMON    CNTMEM(1024)     , ICN    , ICOUNT , IDIN   ,              011220
     1          IDOUT  , IDSCR   , ITELM  , MAINM(1024)     ,              011230
     2          MBUSS  , MEMC    , MEMM   , MICAD(6), MICDA(3),            011240
     3          MICQ   , MSCPAD(16)       , MSR1   , MSR2   ,              011250
     4          NRUN(20), FINIS  , ERROR  , NINSTR , NOPND  ,              011260
     5          MICRO(25),MCNT   , STIME  , ITIME  , IETIM                 011270
      COMMON    KAD    , KDA    , IOLOB                                     011280
      DATA LHIGH /1777778/                                                  011290
      IF (NRUN(3).EQ.2) WRITE (7, 5)                                        011300
    5 FORMAT (27H0SUBROUTINE  SUM  ENTERED.  )                              011310
      MOD = MICRO(2)                                                        011320
      IAR = MICRO(3) + 1                                                    011330
      IBR = MICRO(4) + 1                                                    011340
      IOLOB = MSCPAD(IBR)                                                   011350
      IF (IOP.EQ.23)                                                        011360
         THEN                                                               011370
            ITMP = MSCPAD(IAR) + MSCPAD(IBR) + ICN                          011380
      OR IF (IOP.EQ.24)                                                     011390
         THEN                                                               011400
            ITMP = IDIN + MSCPAD(IBR) + ICN                                 011410
      OR IF (IOP.EQ.25)                                                     011420
         THEN                                                               011430
            ITMP = MSCPAD(IAR) + MICQ + ICN                                 011440
      OR IF (IOP.EQ.26)                                                     011450
         THEN                                                               011460
            ITMP = IDIN + MICQ + ICN                                        011470
      END IF                                                                011480
      ITMP = AND (ITMP, LHIGH)                                              011490
      IF (NRUN(3).EQ.2)                                                     011500
         THEN                                                               011510
            WRITE (7, 25) IOP, ITMP                                         011520
   25       FORMAT ( 7H0INSTR , 02, 24H EXECUTED, TEMP VALUE =  ,06,  14.)  011530
      END IF                                                                011540
      IF (MOD.EQ.7)                                                         011550
         THEN                                                               011560
            MICQ = ITMP                                                     011570
            IDOUT = ITMP                                                    011580
         ELSE                                                               011590
            MSCPAD(IBR) = ITMP                                              011600
            CALL INMOD (IAR, IBR)                                           011610
      END IF                                                                011620
      RETURN                                                                011630
      END                                                                   011640

0 ERRORS FOUND IN THIS ROUTINE.    52 CARDS READ.          56 CARDS OUTPUT.
```

```
      SUBROUTINE DIFF (IOP)                                                 011650
C                                                                           011660
C***** ********************************************************************011670
C                                                                           *011680
C     ALU SIMULATION FOR OP CODES 27 THRU 31                                *011690
C                                                                           *011700
C***** ********************************************************************011710
C                                                                           011720
      LOGICAL   FINIS  , ERROR                                              011730
      COMMON    CNTMEM(1024)     , IGN    , ICOUNT , IDIN   ,              011740
     1          IDOUT  , IDSCR  , ITELM  , MAINM(1024)     ,              011750
     2          MBUSS  , MEMC   , MEMM   , MICAD(6), MICDA(3),            011760
     3          MICQ   , MSCPAD(16)      , MSR1   , MSR2   ,              011770
     4          NRUN(20), FINIS , ERROR  , MINSTR , MOPND  ,              011780
     5          MICRO(25),MCNT   , STIME  , ITIME  , IETIM                011790
      COMMON    KAD    , KDA    , IOLD8                                    011800
      DATA LHIGH /1777778/                                                  011810
      IF (NRUN(3).EQ.2) WRITE (7, 5)                                        011820
    5 FORMAT (27H0SUBROUTINE DIFF  ENTERED.  )                              011830
      MOD = MICRO(2)                                                        011840
      IAR = MICRO(3) + 1                                                    011850
      IBR = MICRO(4) + 1                                                    011860
      ICLD3 = MSCPAD(IBR)                                                   011870
      IF (IOP.EQ.27)                                                        011880
         THEN                                                                011890
            ITMP = MSCPAD(IAR) + AND(COMPL(MSCPAD(IBR)), LHIGH) + IGN        011900
      OR IF (IOP.EQ.28)                                                     011910
         THEN                                                                011920
            ITMP = MSCPAD(IBR) + AND(COMPL(MSCPAD(IAR)), LHIGH) + IGN        011930
      OR IF (IOP.EQ.29)                                                     011940
         THEN                                                                011950
            ITMP = IDIN + AND(COMPL(MSCPAD(IBR)), LHIGH) + IGN              011960
      OR IF (IOP.EQ.30)                                                     011970
         THEN                                                                011980
            ITMP = MSCPAD(IBR) + AND(COMPL(IDIN), LHIGH) + IGN              011990
      OR IF (IOP.EQ.31)                                                     012000
         THEN                                                                012010
            ITMP = IDIN + AND(COMPL(MICQ), LHIGH) + IGN                     012020
      END IF                                                                012030
      ITMP = AND (ITMP, LHIGH)                                              012040
      IF (NRUN(3).EQ.2)                                                     012050
         THEN                                                                012060
            WRITE (7, 25) IOP, ITMP                                         012070
   25       FORMAT ( 7HBINSTR , 02, 24H EXECUTED, TEMP VALUE =  ,06,  1H.)  012080
      END IF                                                                012090
      IF (MOD.EQ.7)                                                         012100
         THEN                                                                012110
            MICQ = ITMP                                                     012120
            IDOUT = ITMP                                                    012130
         ELSE                                                               012140
            MSCPAD(IBR) = ITMP                                              012150
            CALL INMOD(IAR, IBR)                                            012160
      END IF                                                                012170
      RETURN                                                                012180
      END                                                                   012190
```

0 ERRORS FOUND IN THIS ROUTINE. 55 CARDS READ. 61 CARDS OUTPUT.

```
      SUBROUTINE PRBIN                                                  012200
C                                                                       012210
C***** ************************************************************** *012220
C                                                                       *012230
C     EXECUTE THOSE CONTROL FUNCTIONS REQUIRED PRIOR TO                 *012240
C     INSTRUCTION INTERPRETATION AND EXECUTION                          *012250
C                                                                       *012260
C***** ************************************************************** *012270
C                                                                       012280
      LOGICAL   FINIS  ,  ERROR                                         012290
      COMMON    CNTMEM(1024)     ,  ICN    ,  ICOUNT ,  IDIN   ,      012300
     1          IDOUT  ,  IDSCR  ,  ITELM  ,  MAINM(1024)      ,      012310
     2          MBUSS  ,  MEMC   ,  MEMM   ,  MICAD(6), MICDA(3),     012320
     3          MICQ   ,  MSCPAD(16)       ,  MSR1   ,  MSR2   ,      012330
     4          NRUN(20), FINIS  ,  ERROR  ,  MINSTR ,  MOPND  ,      012340
     5          MICRO(25),MCNT   ,  STIME  ,  ITIME  ,  IETIM         012350
      COMMON    KAD    ,  KDA    ,  IOLOB                             012360
      IF (MICRO(6).EQ.1) MSR1 = 2                                       012370
      IF (MICRO(10).EQ.1)                                               012380
        THEN                                                            012390
          MBUSS = MAINM(MEMC + 1)                                       012400
          IDIN = MBUSS                                                  012410
      END IF                                                            012420
      IF (MICRO(13).EQ.1)                                               012430
        THEN                                                            012440
          MBUSS = NOT (OR (MICAD(KAD), IDSCR) )                         012450
          IDIN = MBUSS                                                  012460
      END IF                                                            012470
      IF (MICRO(15).EQ.1) MBUSS = OR (MBUSS, MSR1)                      012480
      ICN = MICRO(17)                                                   012490
      IF (MICRO(20).EQ.1)                                               012500
        THEN                                                            012510
          KDA = 1                                                       012520
      OR IF (MICRO(21).EQ.1)                                            012530
        THEN                                                            012540
          KDA = 2                                                       012550
      OR IF (MICRO(22).EQ.1)                                            012560
        THEN                                                            012570
          KDA = 3                                                       012580
        ELSE                                                            012590
          KDA = 0                                                       012600
      END IF                                                            012610
      IF (MICRO(23).EQ.1)                                               012620
        THEN                                                            012630
          KAD = MICRO(25)                                               012640
          IF (KAD.EQ.0) KAD = 3                                         012650
      END IF                                                            012660
      RETURN                                                            012670
      END                                                               012680
```

0 ERRORS FOUND IN THIS ROUTINE. 43 CARDS READ. 50 CARDS OUTPUT.

```
      SUBROUTINE BINRY                                                  012690
C                                                                       012700
C***** ********************************************************        012710
C                                                                      *012720
C     EXECUTE CONTROL FUNCTIONS SPECIFIED BY BINARY BITS 27-45         *012730
C                                                                      *012740
C***** ********************************************************        012750
C                                                                       012760
      LOGICAL   FINIS  , ERROR                                          012770
      COMMON    CNTMEM(1024)     , IGN    , ICOUNT , IOIN   ,          012780
     1          IDOUT  , IDSCR   , ITELM  , MAINM(1024)     ,          012790
     2          MBUSS  , MEMC    , MEMM   , MICAD(6), MICDA(3),         012800
     3          MICQ   , MSCPAD(16)       , MSR1   , MSR2   ,          012810
     4          NRUN(20), FINIS  , ERROR  , NINSTR , MOPND  ,          012820
     5          MICRO(25),MCNT   , STIME. , ITIME  , IETIM             012830
      COMMON    KBD    , KDA    , IOLOB                                 012840
      DATA MSK10 /#7778/                                                012850
      DATA IOFF /14088/                                                 012860
      IF (MICRO(11).EQ.1) MBUSS = IDOUT                                 012870
      IF (MICRO(7).EQ.1) MSR2 = MSR1                                    012880
      IF (MICRO(8).EQ.1)                                                012890
        THEN                                                               012900
          MEMM = MICRO(5)                                                  012910
        ELSE                                                            012920
          IF (NRUN(3).EQ.2)  WRITE (7, 4) MEMM                             012930
    4     FORMAT (31H M-REG LOAD BIT WAS SET AT LOC , O4,15H - RUN HALTED. )012940
          MEMM = 0                                                         012950
          MICRO(5) = 0                                                     012960
          ERROR = .TRUE.                                                   012970
      END IF                                                            012980
      IF (MICRO(9).EQ.1)                                                012990
        THEN                                                               013000
          M = MEMC + 1                                                     013010
          IF (NRUN(3).EQ.2)                                                013020
            THEN                                                              013030
              WRITE (7, 5) MEMC, MAINM(M), MBUSS                                 013040
    5         FORMAT (17H0MAIN MEMORY LOC , O4, 14H CHANGED FROM , O6, 4H TO ,  013050
     .                06, 19H BY CONTROL BIT 35.   )                          013060
          END IF                                                           013070
          MAINM(M) = MBUSS                                                 013080
      END IF                                                            013090
      IF (MICRO(12).EQ.1.AND.MEMM.NE.0)                                 013100
        THEN                                                               013110
          WRITE (7, 10) MEMM                                               013120
   10     FORMAT (34H0ERROR: FETCH BIT ON BUT M-ADDR =  , O6, 1H. )        013130
          ERROR = .TRUE.                                                   013140
      END IF                                                            013150
      IF (MICRO(14).EQ.1)                                               013160
C       CHECK ALU OUTPUT; IF = ZERO, SET SR1 = 1:                          013170
        THEN                                                               013180
          IF (IDOUT.EQ.0) MSR1 = 1                                         013190
      END IF                                                            013200
      IF (MICRO(16).EQ.1)                                               013210
        THEN                                                               013220
          MEMM = OR (MICRO(5), MSR1)                                       013230
          IF (NRUN(3).EQ.2) WRITE (7, 3) MEMM                              013240
```

```
   30         FORMAT (29HONEXT H MODIFIED BY MSR1 TO:    , 04 )                  013250
      END IF                                                                    013260
C     IF BIT 47 ON, BOTTOM 10 BITS OF BUSS TO C REGISTER:                       013270
      IF (MICRO(19).EQ.1) MEMC = AND (MBUSS, MSK10)                             013280
      IF (MICRO(18).EQ.0) ITELM = MBUSS                                         013290
      IF (KDA.NE.0)                                                             013300
         THEN                                                                   013310
           MICDA(KDA) = MBUSS                                                   013320
           KDA = 0                                                              013330
      END IF                                                                    013340
      IF (NRUN(3).EQ.2) CALL REPT1                                              013350
      RETURN                                                                    013360
      END                                                                       013370
```

0 ERRORS FOUND IN THIS ROUTINE. 63 CARDS READ. 68 CARDS OUTPUT.

```
      SUBROUTINE STATS                                                      013380
C                                                                           013390
C***** ***************************************************************** *013400
C                                                                         *013410
C     OUTPUT FINAL CONTENTS AND STATISTICS                                *013420
C                                                                         *013430
C***** ***************************************************************** *013440
C                                                                           013450
      LOGICAL   FINIS  ,  ERROR                                             013460
      COMMON    CNTMEM(1024)     ,  ICN    ,  ICOUNT ,  IDIN   ,          013470
     1          IDOUT  ,  IDSCR  ,  ITELM  ,  MAINM(1024)      ,          013480
     2          MBUSS  ,  MEMC   ,  MEMH   ,  MICAD(6), MICDA(3),         013490
     3          MICQ   ,  MSCPAD(16)       ,  MSR1   ,  MSR2   ,         013500
     4          NRUN(20), FINIS  ,  ERROR  ,  MINSTR ,  MOPND  ,          013510
     5          MICRO(25),MCNT   ,  STIME  ,  ITIME  ,  IETIM             013520
      COMMON    KAD    ,  KDA    ,  IOLD3                                   013530
      WRITE (7, 10) STIME, ICOUNT, MCNT, MSCPAD, IDIN, IDOUT, MBUSS,       013540
     .              MEMC, MEMH, MICQ, MICAD, MICDA, ICN,IDSCR,MSR1,MSR2    013550
   10 FORMAT (26H1RUN TERMINATED AT TIME :  ,F15.8,15H SECONDS AFTER   ,    013560
     .        I5,11H MICROS OR , I5,17H MACROS EXECUTED.    //             013570
     .        21H0SCRATCHPAD CONTAINS:,/1H , 16( /5X, 08), //             013580
     .        11H0DATA IN = , 08,13H, DATA OUT = , 08, 9H, BUSS = , 08,   013590
     .        10H, C-REG = ,04,10H, H-REG = ,04,10H, Q-REG = , 06, //     013600
     .        9H0A/D S = ,5(2X,08),10H, D/A S = ,3(2X,08),//              013610
     .        11H0 CARRY =   ,I2, 20H, INPUT DISCRETES =  , 08, //        013620
     .        9H0 MSR1 = , I1, 9H, MSR2 =  , I1 )                         013630
      RETURN                                                                013640
      END                                                                   013650

     0 ERRORS FOUND IN THIS ROUTINE.    28 CARDS READ.          21 CARDS OUTPUT.
```

```
      SUBROUTINE REPT1                                                  013660
C                                                                       013670
C***** ********************************************************** ****013680
C                                                                      *013690
C     DETAILED OUTPUT FROM MICRO-INSTRUCTION EXECUTION                 *013700
C                                                                      *013710
C***** ********************************************************** ****013720
C                                                                       013730
      LOGICAL   FINIS  ,  ERROR                                         013740
      COMMON    CNTMEM(1024)     ,  IGN    ,  ICOUNT ,  IDIN   ,      013750
     1          IDOUT  ,  IDSCR  ,  ITELM  ,  MAINM(1024)      ,      013760
     2          MBUSS  ,  MEMC   ,  MEMH   ,  MICAD(6), MICDA(3),     013770
     3          MICQ   ,  MSCPAD(16)        ,  MSR1   ,  MSR2   ,     013780
     4          NRUN(20), FINIS  ,  ERROR  ,  NINSTR ,  NOPND  ,      013790
     5          MICRO(25),MCNT   ,  STIME  ,  ITIME  ,  IETIM         013800
      COMMON    KAD    ,  KDA    ,  IOLOB                               013810
      INSTR = MICRO(1)                                                  013820
      MOD = MICRO(2)                                                    013830
      IA = MICRO(3) + 1                                                 013840
      IB = MICRO(4) + 1                                                 013850
      I = IA - 1                                                        013860
      J = IB - 1                                                        013870
      WRITE (7,10)INSTR,MOD,I,MSCPAD(IA),J,IOLOB,MSCPAD(IB),MICQ2,MICQ  013880
     1          , ITEL2, ITELM                                         013890
   10 FORMAT (18H0EXECUTED OP CODE ,O2,13H WITH MODIFIER ,I1,1H. //     013900
     .        7H A-REG ,O2,12H CONTENTS =  ,O5,// 7H B-REG ,O2,        013910
     .        22H CONTENTS INITIALLY =  ,O5, 8H, NOW =  ,O6, //        013920
     .        28H Q-REG CONTENTS INITIALLY =  ,O6,8H, NOW =  ,O6, //   013930
     .        28H TELEM CONTENTS INITIALLY =  ,O6,8H, NOW =  ,O6 // )  013940
      ITEL2 = ITELM                                                     013950
      MICQ2 = MICQ                                                      013960
      RETURN                                                            013970
      END                                                               013980

     0 ERRORS FOUND IN THIS ROUTINE.    33 CARDS READ.          26 CARDS OUTPUT.


      SUBROUTINE REPT2                                                  013990
C     DETAILED OUTPUT FROM INSTRUCTION MODIFICATION AND CONTROLS        014000
      RETURN                                                            014010
      END                                                               014020

     0 ERRORS FOUND IN THIS ROUTINE.     4 CARDS READ.           3 CARDS OUTPUT.

     0 PREPROCESSOR ERRORS FOUND.    1392 TOTAL CARDS READ.    1387 TOTAL CARDS OUTPUT.
```

```
      PROGRAM EMUL      74/74   OPT=1                    FTN 4.2+74355      05/11/75  10.03.03.

                        PROGRAM EMUL (INPUT, OUTPUT, TAPE5 = INPUT, TAPE7=OUTPUT)                  1
                        LOGICAL FINIS, ERROR                                                       2
                        COMMON    CNTMEM(1024)     , ICN    , ICOUNT ,  IOIN    ,               3
                       1          IDOUT  ,  IDSCR  ,  ITELM  ,  MAINM(1024)    ,               4
                       2          MBUSS  ,  MEMC   ,  MEMH   ,  MICAD(6), MICDA(3),            5
                       3          MICQ   ,  MSCPAD(16)       ,  MSR1   ,  MSR2   ,            6
                       4          NRUN(20), FINIS  ,  ERROR  ,  MINSTR ,  MOPND  ,            7
                       5          MICRO(25),MCNT   ,  STIME  ,  ITIME  ,  IETIM               8
                        COMMON    KAD    ,  KDA    ,  IOLDB                                     9
                        DATA LFTMS /3778/                                                         10
                        READ (5, 10) NRUN                                                         11
                     10 FORMAT (20 O 4)                                                           12
                        WRITE (7, 20) NRUN                                                        13
                     20 FORMAT (20H1RUN CONTROL CARD =    , 20O4 )                             14
                        TINCR = 0.00000375                                                        15
                        CALL INIT                                                                 16
                        CALL SETCM                                                                17
                        CALL SETMM                                                                18
                        CALL LOADM                                                                19
                        MICS = 0                                                                  20
                        IF(.NOT.                                                                  21
                      .   (NRUN(4).EQ.1)                                                          22
                      .)GO TO  99999                                                              23
                        N = 1                                                                     24
                        WRITE (7, 80)                                                             25
                     80 FORMAT (10H1 LOCATION,5X,15HMAIN   DECODED,26X,8HLOCATION,5X,             26
                      .        15HMAIN   DECODED, /11H  DEC   OCT,3X,12HMEMORY LEFT ,             27
                      .         5HRIGHT,4X,14HCONTROL MEMORY,7X,9HDEC   OCT ,3X,                28
                      .        17HMEMORY LEFT RIGHT, 4X,14HCONTROL MEMORY //  )                  29
                  99998 IF(.NOT.                                                                  30
                      .          (N.LE.1.24)                                                      31
                      .)GO TO 99997                                                               32
                        NN = N - 1                                                                33
                        NR = N + 1                                                                34
                        MAIN = MAINM(N)                                                           35
                        MAINL = SHIFT (MAIN, -8)                                                  36
                        MAINR = AND (MAIN, LFTMS)                                                 37
                        MAIN = MAINM(NR)                                                          38
                        MAINJ = SHIFT(MAIN, -8)                                                   39
                        MAINS = AND (MAIN, LFTMS)                                                 40
                        WRITE (7, 90)NN,NN,MAINM(N),MAINL,MAINR,CNTMEM(N),                        41
                      .              N,N,MAINM(NR),MAINJ,MAINS,CNTMEM(NR)                         42
                     90 FORMAT (1H ,I4,2X,O4,3X,O6,2X,O3,2X,O3,3X,O16,5X,                         43
                      .              I4,2X,O4,3X,O6,2X,O3,2X,O3,3X,O16    )                     44
                        N = N + 2                                                                 45
                        GO TO 99998                                                               46
                  99997 CONTINUE                                                                  47
                  99999 CONTINUE                                                                  48
                  99996 IF(.NOT.                                                                  49
                      .          (.NOT.(FINIS.OR.ERROR))                                          50
                      .)GO TO 99995                                                               51
                        CALL DCOOM                                                                52
                  99994 IF(.NOT.                                                                  53
                      .          (MEMH.NE.0)                                                      54
                      .)GO TO 99993                                                               55
                        MHOLD = MEMH                                                              56
                        CALL DCOOC                                                                57
```

```
                      CALL EXECM                                                  58
                      IF(.NOT.                                                    59
     60               .   (NRUN(3).EQ.2)                                          60
                      .)GO TO  99992                                              61
                      N = MMOLD + 1                                               62
                      WRITE (7, 15) MMOLD, CNTMEM(N), (MICRO(I), I = 1, 25)       63
                   15 FORMAT (53HOTHE FOLLOWING CM INSTRUCTION HAS JUST BEEN EXECUTED: ,  64
     65               .     //1H , O4, 5X,O16,4(3X,O2),4H  M: ,O4,2X, 20(1X,I1)///)  65
                      M = 1                                                       66
                   99991 IF(.NOT.                                                 67
                      .         (M.LE.15)                                       68
                      .)GO TO 99990                                               69
     70               K = M - 1                                                   70
                      WRITE (7, 30) K, MSCPAD(M)                                  71
                   30 FORMAT (13H SCRATCH PAD( , O2, 11H) CONTAINS  , O6,  2H .)  72
                      M = M + 1                                                   73
                      GO TO 99991                                                 74
     75            99990 CONTINUE                                                75
                      WRITE (7, 50) IDIN, IDOUT, NBUSS, NEMC, ICN, MSR1, MSR2     76
                   50 FORMAT (11HODATA IN = ,O8,13H, DATA OUT = ,O8,9H, BUSS = ,O8,  77
                     .       10H, C-REG = ,O8,/9H CARRY = ,I1,8H, SR1 = ,I1,9H, SR2 = ,  78
                     .       I1, 1H.)                                             79
     80            99992 CONTINUE                                                80
                      STIME = STIME + TINCR                                       81
                      ITIME = IFIX(STIME)                                         82
                      MICS = MICS + 1                                             83
                      IF(.NOT.                                                    84
     85               .   (NRUN(3).EQ.2)                                          85
                      .)GO TO  99989                                              86
                      WRITE (7, 40) STIME, ITIME, MICS                            87
                   40 FORMAT (1HO, F15.8, 5X, I8, 5X, I5)                         88
                   99989 CONTINUE                                                 89
     90               IF(.NOT.                                                    90
                      .   (NRUN(9).EQ.5)                                          91
                      .)GO TO  99988                                              92
                      IF(.NOT.                                                    93
                      .   (MICS.GE.NRUN(10)))                                     94
     95               .)GO TO  99987                                              95
                      MEMM = 0                                                    96
                      FINIS = .TRUE.                                              97
                   99987 CONTINUE                                                 98
                   99988 CONTINUE                                                 99
    100               GO TO 99994                                                 100
                   99993 CONTINUE                                                 101
                      MCNT = MCNT + 1                                             102
                      CALL FINAL                                                  103
                      GO TO 99996                                                 104
    105            99995 CONTINUE                                                105
                      IF(.NOT.                                                    106
                      .   (NRUN(4).EQ.1)                                          107
                      .)GO TO  99986                                              108
                      N = 1                                                       109
    110               WRITE (7,80)                                                110
                   99985 IF(.NOT.                                                 111
                      .         (N.LE.1;24)                                       112
                      .)GO TO 99984                                               113
                      NN = N - 1                                                  114
```

```
      PROGRAM EMUL       74/74   OPT=1                    FTN 4.2+74355      05/11/75  10.03.03.

  115                 NR = N + 1                                                          115
                      MAIN = MAINM(N)                                                     116
                      MAINL = SHIFT (MAIN, -8)                                            117
                      MAINR = AND (MAIN, LFTMS)                                           118
                      MAIN = MAINM(NR)                                                    119
  120                 MAINJ = SHIFT (MAIN, -8)                                            120
                      MAINS = AND (MAIN, LFTMS)                                           121
                      WRITE (7, 90) NN,NN,MAINM(N),MAINL,MAINR,CNTMEM(N),                 122
                     .                N,N,MAINM(NR),MAINJ,MAINS,CNTMEM(NR)              123
                                                                                          124
                      N = N + 2
  125                 GO TO 99985                                                         125
            99984 CONTINUE                                                            126
            99986 CONTINUE                                                            127
                      CALL STATS                                                          128
                      STOP                                                                129
  130                 END                                                                 130
```

```
      SUBROUTINE INIT                                                   131
      LOGICAL   FINIS , ERROR                                           132
      COMMON    CNTMEM(1024)     , ICN    , ICOUNT , IDIN   ,          133
     1          IOOUT  , IDSCR  , ITELM  , MAINM(1024)     ,          134
     2          MBUSS  , MEMC   , MEMH   , MICAD(6), MICDA(3),        135
     3          MICQ   , MSCPAD(16)      , MSR1   , MSR2   ,          136
     4          NRUN(2), FINIS  , ERROR  , MINSTR , MOPND  ,          137
     5          MICRO(25),MCNT  , STIME  , ITIME  , IETIM             138
      COMMON    KAD    , KDA    , IOLOB                               139
      ICN = 0                                                           140
      ICOUNT = 0                                                        141
      IDIN = 0                                                          142
      IOOUT = 0                                                         143
      IDSCR = 0                                                         144
      ITELM = 0                                                         145
      KAD = 0                                                           146
      KDA = 0                                                           147
      MBUSS = 0                                                         148
      MEMC = 0                                                          149
      MEMH = 0                                                          150
      MICQ = 0                                                          151
      MSR1 = 0                                                          152
      MSR2 = 0                                                          153
      MSCPAD(1) = NRUN(5)                                               154
      N = 2                                                             155
99999 IF(.NOT.                                                          156
     .          (N.LE.15)                                              157
     .)GO TO 99998                                                      158
      MSCPAD(N) = 0                                                     159
      N = N + 1                                                         160
      GO TO 99999                                                       161
99998 CONTINUE                                                          162
      N = 1                                                             163
99997 IF(.NOT.                                                          164
     .          (N.LE.3)                                               165
     .)GO TO 99996                                                      166
      MICAD(N) = 0                                                      167
      N = N + 1                                                         168
      GO TO 99997                                                       169
99996 CONTINUE                                                          170
      MICDA(1) = 0                                                      171
      MICDA(2) = 0                                                      172
      MICDA(3) = 0                                                      173
      FINIS = .FALSE.                                                   174
      ERROR = .FALSE.                                                   175
      MCNT = 0                                                          176
      STIME = 0.0                                                       177
      ITIME = 0                                                         178
      IETIM = 0                                                         179
      RETURN                                                            180
      END                                                               181
```

```
      SUBROUTINE DCODC
      LOGICAL   FINIS  ,  ERROR
      COMMON    CNTMEM(1024)     ,  ICN    ,  ICOUNT ,  IDIN   ,
     1          IDOUT  ,  IDSCR  ,  ITELM  ,  MAINM(1024)     ,
     2          MBUSS  ,  MEMC   ,  MEMH   ,  MICAD(6), MICDA(3),
     3          MICQ   ,  MSCPAD(16)        ,  MSR1   ,  MSR2   ,
     4          NRUN(3), FINIS  ,  ERROR  ,  MINSTR ,  MOPNO  ,
     5          MICRO(25),MCNT   ,  STIME  ,  ITIME  ,  IETIM
      COMMON    KAD    ,  KDA    ,  IOLOB
      DATA MSK1 /68/
      IPTR = MEMH
      IERR = 0
      CALL GETCM (IPTR, 3, 5, MOP, IFLAG)
      MICRO(1) = XOR (MSK1, MOP)
      IERR = IFLAG
      CALL GETCM (IPTR, 5, 3, MOP, IFLAG)
      MICRO(2) = MOP
      IERR = OR (IERR, IFLAG)
      CALL GETCM (IPTR, 8, 4, MOP, IFLAG)
      MICRO(3) = MOP
      IERR = OR(IERR, IFLAG)
      CALL GETCM (IPTR, 12, 4, MOP, IFLAG)
      MICRO(4) = MOP
      IERR = OR(IERR, IFLAG)
      CALL GETCM (IPTR, 16, 10, MOP, IFLAG)
      MICRO(5) = MOP
      IERR = OR(IERR, IFLAG)
      CALL GETCM (IPTR, 45, 3, MOP, IFLAG)
      MICRO(25) = MOP
      IERR = OR(IERR, IFLAG)
      IFLAG = IERR
      NCOL = 26
      INDX = 6
99999 IF(.NOT.
     .       (IFLAG.EQ.0.AND.INDX.LE.24)
     .)GO TO 99998
      CALL GETCM (IPTR, NCOL, 1, MOP, IFLAG)
      MICRO(INDX) = MOP
      NCOL = NCOL + 1
      INDX = INDX + 1
      GO TO 99999
99998 CONTINUE
      IF(.NOT.
     .   (IFLAG.EQ.0)
     .)GO TO  99997
      IF(.NOT.
     .   (NRUN(3).EQ.2)
     .)GO TO  99996
      WRITE (7, 5) MEMH, MICRO(5)
    5 FORMAT (     15HLTHIS INSTR AT: , O4, 12H, H-FIELD = , O4, 2H . )
99996 CONTINUE
      GO TO 99995
99997 CONTINUE
      WRITE (7, 10) IFLAG
   10 FORMAT (43HOINVALID CALL TO GETCM FROM DCODC, IFLAG =  ,I2,1H. )
      ERROR = .TRUE.
99995 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE INMOD (IA, IB)                                                 241
      LOGICAL   FINIS  ,  ERROR                                                 242
      COMMON    CNTME(1024)     ,  ICN    ,  ICOUNT ,  IDIN   ,               243
     1          IDOUT  ,  IOSCR  ,  ITELM  ,  MAINM(1024)      ,               244
  5   2          MBUSS  ,  MEMC   ,  MEMH   ,  MICAD(6), MICDA(3),             245
      3          MICQ   ,  MSCPAD(16)       ,  MSR1   ,  MSR2   ,             246
      4          NRUN(2),  FINIS  ,  ERROR  ,  MINSTR ,  MOPND  ,             247
      5          MICRO(25),MCNT   ,  STIME  ,  ITIME  ,  IETIM                248
      COMMON    KAD    ,  KDA    ,  IOLOB                                     249
 10   DATA KMASK /1777778/                                                     250
      DATA IMIBIT/1000000B/                                                    251
      IOP = MICRO(1)                                                           252
      MOD = MICRO(2)                                                           253
      IF(.NOT.                                                                 254
 15  .   (MOD.EQ.0)                                                           255
     .)GO TO  99999                                                            256
      IDOUT = MSCPAD(I3)                                                       257
      GO TO 99998                                                              258
99999 IF(.NOT.                                                                 259
 20  .      (MOD.EQ.1)                                                        260
     .)GO TO 99997                                                             261
      IDOUT = MSCPAD(I4)                                                       262
      GO TO 99998                                                              263
99997 IF(.NOT.                                                                 264
 25  .      (MOD.EQ.2)                                                        265
     .)GO TO 99996                                                             266
      IDOUT = IOLOB                                                            267
      GO TO 99998                                                              268
99996 IF(.NOT.                                                                 269
 30  .      (MOD.EQ.3)                                                        270
     .)GO TO 99995                                                             271
      IBIT = SHIFT (MSCPAD(I3), -15)                                           272
      IF(.NOT.                                                                 273
     .   (IBIT.EQ.0)                                                           274
 35  .)GO TO  99994                                                            275
      MSR1 = 1                                                                 276
      GO TO 99993                                                              277
99994 CONTINUE                                                                 278
      MSR1 = 0                                                                 279
 40   99993 CONTINUE                                                           280
      MSCPAD(I3) = AND ((SHIFT(MSCPAD(I3),1)), KMASK)                          281
      IDOUT = MSCPAD(I3)                                                       282
      GO TO 99998                                                              283
99995 IF(.NOT.                                                                 284
 45  .      (MOD.EQ.4)                                                        285
     .)GO TO 99992                                                             286
      MSCPAD(I3) = SHIFT(MSCPAD(I3), -1)                                       287
      MSCPAD(I3) = OR(MSCPAD(I3), IMIBIT)                                      288
      IDOUT = MSCPAD(I3)                                                       289
 50   GO TO 99998                                                              290
99992 IF(.NOT.                                                                 291
     .      (MOD.EQ.5)                                                        292
     .)GO TO 99991                                                             293
      IBIT = SHIFT (MSCPAD(I3), -15)                                           294
 55   IF(.NOT.                                                                 295
     .   (IBIT.EQ.0)                                                           296
     .)GO TO  99990                                                            297
```

```
                      MSR1 = 1                                                  298
                      GO TO 99989                                               299
60          99990     CONTINUE                                                  300
                      MSR1 = 0                                                  301
                99989 CONTINUE                                                  302
                      IBIT = SHIFT (MICQ, -15)                                  303
                      MICQ = AND ( SHIFT(MICQ, 1) , KMASK )                     304
65                    MICQ = OR(MICQ, 1)                                        305
                      MSCPAD(IB) = AND (SHIFT(MSCPAD(IB), 1) , KMASK)           306
                      MSCPAD(IB) = OR (MSCPAD(IB), IBIT)                        307
                      IDOUT = MSCPAD(IB)                                        308
                      GO TO 99998                                               309
70          99991     IF(.NOT.                                                  310
                     .      (MOD.EQ.6)                                         311
                     .)GO TO 99988                                             312
                      IBIT = AND (MSCPAD(IB), 1)                                313
                      IBIT = SHIFT (IBIT, 15)                                   314
75                    LOBIT = AND (MICQ, 1)                                     315
                      IF(.NOT.                                                  316
                     .   (LOBIT.EQ.0)                                           317
                     .)GO TO  99987                                             318
                      MSR1 = 1                                                  319
80                    GO TO 99986                                               320
                99987 CONTINUE                                                  321
                      MSR1 = 0                                                  322
                99986 CONTINUE                                                  323
                      MICQ = SHIFT (MICQ, -1)                                   324
85                    MICQ = OR(MICQ, IBIT)                                     325
                      MSCPAD(IB) = SHIFT (MSCPAD(IB), -1)                       326
                      MSCPAD(IB) = OR(MSCPAD(IB), IHIBIT)                       327
                      IDOUT = MSCPAD(IB)                                        328
                      GO TO 99998                                               329
90          99988     CONTINUE                                                  330
                      WRITE (7, 5) MOD                                          331
                    5 FORMAT (26H INVALID INSTR MODIFIER =   ,O4,15H.  RUN HALTED.  )  332
                      ERROR = .TRUE.                                            333
                99998 CONTINUE                                                  334
95                    IF (NRUN(3).EQ.2) WRITE (7, 10) MOD, IDOUT                335
                   10 FORMAT (13HOINSTR MOD =  ,I1,21H JUST SET DATA-OUT =  ,O8, 1H.)  336
                      IF(.NOT.                                                  337
                     .   (NRUN(3).EQ.2.OR.NRUN(3).EQ.3)                         338
                     .)GO TO  99985                                             339
100                   CALL REPT2                                                340
                99985 CONTINUE                                                  341
                      RETURN                                                    342
                      END                                                       343
```

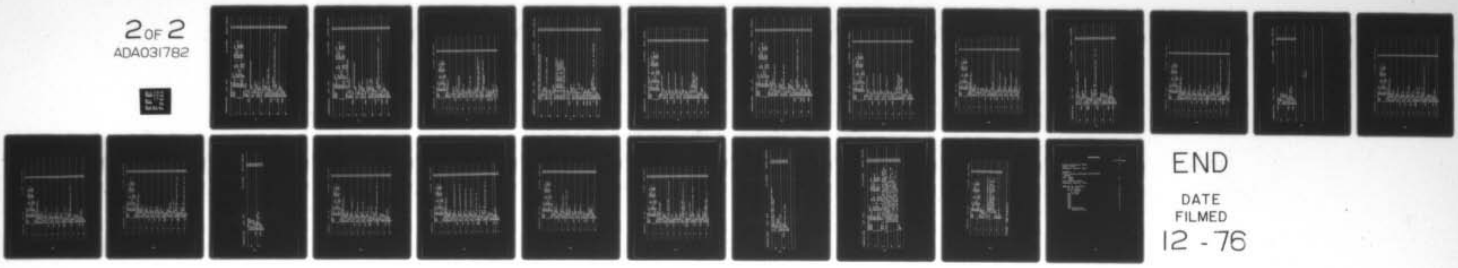
```
      SUBROUTINE EXECM                                                  344
      LOGICAL   FINIS  ,  ERROR                                         345
      COMMON    CNTMEM(1024)    ,  ICN    ,  ICOUNT ,  IDIN   ,       346
     1          IDOUT  ,  IDSCR   ,  ITELM  ,  MAINM(1024)    ,       347
     2          MBUSS  ,  MEMC    ,  MEMM   ,  MICAD(6), MICDA(3),    348
     3          MICQ   ,  MSCPAD(16)        ,  MSR1   ,  MSR2   ,      349
     4          NRUN(2), FINIS  ,  ERROR  ,  MINSTR ,  MOPND  ,       350
     5          MICRO(25),MCNT  ,  STIME  ,  ITIME  ,  IETIM          351
      COMMON    KAD    ,  KDA   ,  IOLDB                               352
      CALL PRBIN                                                        353
      IF(.NOT.                                                          354
     .   (MSR2.EQ.1)                                                    355
     .)GO TO  99999                                                     356
      MICRO(1) = OR (MICRO(1), 6)                                       357
      IF (NRUN(3).EQ.2) WRITE (7, 20) MICRO(1)                          358
   20 FORMAT (15HOOP CODE NOW =  , O2, 16H DUE TO SR2 = 1.   )          359
99999 CONTINUE                                                          360
      IOP = MICRO(1)                                                    361
      IF(.NOT.                                                          362
     .   (IOP.GE.0.AND.IOP.LE.6)                                        363
     .)GO TO  99998                                                     364
      CALL BOLN (IOP)                                                   365
      GO TO 99997                                                       366
99998 IF(.NOT.                                                          367
     .      (IOP.GE.7.AND.IOP.LE.10)                                    368
     .)GO TO 99996                                                      369
      CALL INVRT (IOP)                                                  370
      GO TO 99997                                                       371
99996 IF(.NOT.                                                          372
     .      (IOP.GE.11.AND.IOP.LE.14)                                   373
     .)GO TO 99995                                                      374
      CALL CMPLT (IOP)                                                  375
      GO TO 99997                                                       376
99995 IF(.NOT.                                                          377
     .      (IOP.GE.15.AND.IOP.LE.18)                                   378
     .)GO TO 99994                                                      379
      CALL TRINC (IOP)                                                  380
      GO TO 99997                                                       381
99994 IF(.NOT.                                                          382
     .      (IOP.GE.19.AND.IOP.LE.22)                                   383
     .)GO TO 99993                                                      384
      CALL DECTR (IOP)                                                  385
      GO TO 99997                                                       386
99993 IF(.NOT.                                                          387
     .      (IOP.GE.23.AND.IOP.LE.26)                                   388
     .)GO TO 99992                                                      389
      CALL SUM  (IOP)                                                   390
      GO TO 99997                                                       391
99992 IF(.NOT.                                                          392
     .      (IOP.GE.27 AND.IOP.LE.31)                                   393
     .)GO TO 99991                                                      394
      CALL DIFF (IOP)                                                   395
      GO TO 99997                                                       396
99991 CONTINUE                                                          397
      WRITE (7, 10) IOP                                                 398
   10 FORMAT (18HOINVALID OP CODE: , I2, 22H ENCOUNTERED IN EXECM.  )   399
      ERROR = .TRUE.                                                    410
99997 CONTINUE                                                          411
      CALL @INRY                                                        412
      ICOUNT = ICOUNT + 1                                               413
      RETURN                                                            414
      END                                                               415
```

```
      SUBROUTINE FINAL                                                  436
      LOGICAL   FINIS  ,  ERROR                                         437
      COMMON    CNTMEM(1024)     ,  ICN    ,  ICOUNT ,  IDIN   ,      438
     1          IDOUT  ,  IDSCR  ,  ITELM  ,  MAINM(1024)      ,      439
     2          MBUSS  ,  MEMC   ,  MEMH   ,  MICAD(6), MICDA(3),     440
     3          MICQ   ,  MSCPAD(16)       ,  MSR1   ,  MSR2   ,      441
     4          NRUN(20),  FINIS  ,  ERROR  ,  MINSTR ,  NOPNO  ,     442
     5          MICRO(25),MCNT    ,  STIME  ,  ITIME  ,  IETIM        443
      COMMON    KA0    ,  K04    ,  IOLDB                             444
      NEND = NRUN(9)                                                    445
      NVAL = NRUN(10)                                                   446
      IF(.NOT.                                                          447
     .   (NEND.EQ.1)                                                    448
     .)GO TO  99999                                                     449
      IF(.NOT.                                                          450
     .   (MEMH.EQ.NVAL)                                                 451
     .)GO TO  99998                                                     452
      FINIS = .TRUE.                                                    453
99998 CONTINUE                                                          454
      GO TO 99997                                                       455
99999 IF(.NOT.                                                          456
     .      (NEND.EQ.2)                                                 457
     .)GO TO 99996                                                      458
      IF(.NOT.                                                          459
     .   (MEMC.EQ.NVAL)                                                 460
     .)GO TO  99995                                                     461
      FINIS = .TRUE.                                                    462
99995 CONTINUE                                                          463
      GO TO 99997                                                       464
99996 IF(.NOT.                                                          465
     .      (NEND.EQ.3)                                                 466
     .)GO TO 99994                                                      467
      IF(.NOT.                                                          468
     .   (ITIME.GE.NVAL)                                                469
     .)GO TO  99993                                                     470
      FINIS = .TRUE.                                                    471
99993 CONTINUE                                                          472
      GO TO 99997                                                       473
99994 IF(.NOT.                                                          474
     .      (NEND.EQ.4)                                                 475
     .)GO TO 99992                                                      476
      IF(.NOT.                                                          477
     .   (IETIM.EQ.NVAL)                                                478
     .)GO TO  99991                                                     479
      FINIS = .TRUE.                                                    480
99991 CONTINUE                                                          481
      GO TO 99997                                                       482
99992 IF(.NOT.                                                          483
     .      (NEND.EQ.5)                                                 484
     .)GO TO 99990                                                      485
      IF(.NOT.                                                          486
     .   (MCNT.GE.NVAL)                                                 487
     .)GO TO  99989                                                     488
      FINIS = .TRUE.                                                    489
99989 CONTINUE                                                          490
      GO TO 99997                                                       491
99990 IF(.NOT.                                                          492
     .      (NEND.EQ.6)                                                 493
     .)GO TO 99988                                                      494
      GO TO 99997                                                       495
99988 CONTINUE                                                          496
      WRITE (7, 5) NEND                                                 497
    5 FORMAT (31HOINVALID RUN TERMINATION CODE = , I3, 1H. )            498
      ERROR = .TRUE.                                                    499
99997 CONTINUE                                                          470
      RETURN                                                            471
      END                                                               472
```

AD-A031 782 ARMY MISSILE RESEARCH DEVELOPMENT AND ENGINEERING LAB--ETC F/G 9/2
MULTIPURPOSE DIGITAL MICROPROCESSOR EMULATOR.(U)
MAY 76 J R BROOKSHIRE

UNCLASSIFIED RG-76-62 NL

2 OF 2
ADA031782



END

DATE
FILMED
12 - 76

```
      SUBROUTINE SETMM                                                  473
      LOGICAL    FINIS  , ERROR                                         474
      COMMON     CNTMEM(1024)     , ICN    , ICOUNT , IDIN   ,        475
     1           IDOUT  , IDSCR  , ITELM  , MAINM(1024)     ,        476
     2           MBUSS  , MEMC   , MEMH   , MICAD(6), MICDA(3),      477
     3           MICQ   , MSCPAD(16)      , MSR1   , MSR2   ,        478
     4           NRUN(20), FINIS , ERROR  , MINSTR , MOPWD  ,        479
     5           MICRO(25),MCNT  , STIME  , ITIME  , IETIM           480
      COMMON     KAD    , KDA    , IOLDB                             481
      DATA MSK /3778/                                                 482
      INDX = 1                                                        483
99999 IF(.NOT.                                                        484
     .          (.NOT.ERROR.AND.INDX.LE.1024)                        485
     .)GO TO 99998                                                    486
      IF(.NOT.                                                        487
     .   (NRUN(2).EQ.1)                                               488
     .)GO TO  99997                                                   489
      MAINM(INDX) = 0                                                 490
      GO TO 99996                                                     491
99997 IF(.NOT.                                                        492
     .      (NRUN(2).EQ.2)                                           493
     .)GO TO 99995                                                    494
      NEWDX = INDX + 1                                                495
      MAINM(INDX) = AND (NEWDX, MSK)                                  496
      GO TO 99996                                                     497
99995 CONTINUE                                                        498
      WRITE (7, 10) NRUN(2)                                           499
   10 FORMAT (44HSETMM CALLED WITH INVALID OPTION NRUN(2) =  , I3, 1H.)  500
      ERROR = .TRUE.                                                  501
99996 CONTINUE                                                        502
      INDX = INDX + 1                                                 503
      GO TO 99999                                                     504
99998 CONTINUE                                                        505
      RETURN                                                          506
      END                                                             507
```

```
      SUBROUTINE SETCM                                                  508
      LOGICAL   FINIS  ,  ERROR                                         509
      COMMON    CNTMEM(1024)     ,  ICN    ,  ICOUNT ,  IDIN   ,      510
     1          IDOUT  ,  IDSCR  ,  ITELM  ,  MAINM(1024)      ,      511
     2          MBUSS  ,  MEMC   ,  MEMH   ,  MICAD(6), MICDA(3),     512
     3          MICQ   ,  MSCPAD(16)       ,  MSR1   ,  MSR2   ,      513
     4          NRUN(2),  FINIS  ,  ERROR  ,  MINSTR ,  MOPND  ,      514
     5          MICRO(25),MCNT   ,  STIME  ,  ITIME  ,  IETIM         515
      COMMON    KAD    ,  KDA    ,  IOLOG                             516
      DATA REPL /0B/, REPL2 /53777777620000008/                         517
      DATA ADDR /20000000B/                                             518
      INDX = 1                                                          519
99999 IF(.NOT.                                                          520
     .       (.NOT.ERROR.AND.INDX.LE.1024)                              521
     .)GO TO 99998                                                      522
      IF(.NOT.                                                          523
     .   (NRUN(1).EQ.1)                                                 524
     .)GO TO  99997                                                     525
      CNTMEM(INDX) = REPL                                               526
      GO TO 99996                                                       527
99997 IF(.NOT.                                                          528
     .      (NRUN(1).EQ.2)                                             529
     .)GO TO 99995                                                      530
      REPL2 = REPL2 + ADDR                                              531
      CNTMEM(INDX) = REPL2                                              532
      GO TO 99996                                                       533
99995 CONTINUE                                                          534
      WRITE (7, 10) NRUN(1)                                             535
   10 FORMAT (44HOSETCM CALLED WITH INVALID OPTION NRUN(1) =  , I3,1H. ) 536
      ERROR = .TRUE.                                                    537
99996 CONTINUE                                                          538
      INDX = INDX + 1                                                   539
      GO TO 99999                                                       540
99998 CONTINUE                                                          541
      RETURN                                                            542
      END                                                               543
```

```
      SUBROUTINE LOADM                                                  546
      LOGICAL    FINIS  , ERROR                                         545
      COMMON     CNTMEM(1024)     , ICM    , ICOUNT , IOIM   ,        546
     1           IOOUT  , IOSCR   , ITELM  , MAINM(1024)     ,        547
     2           MBUSS  , MEMC    , MEMM   , MICAD(6), MICDA(3),      548
     3           MICQ   , MSCPAD(16)       , MSR1   , MSR2   ,        549
     4           NRUN(2), PINIS   , ERROR  , MINSTR , MOPMD  ,        550
     5           MICRO(25),MCNT   , STIME  , ITIME  , IETIM           551
      COMMON     KAD    , KDA     , IOLOB                             552
     10 DIMENSION CNTFLD(6)                                           553
      DATA MSK1 /7400000000000000007/                                  554
      KEY = 0                                                          555
      K = 0                                                            556
      L = 0                                                            557
     15 IF(.NOT.                                                       558
     .   (NRUN(11).EQ.1)                                              559
     .)GO TO 99999                                                    560
   99998 IF(.NOT.                                                     561
     .       (KEY.NE.9)                                               562
     .)GO TO 99997                                                    563
     20 READ (5, 10) KEY, NADR, MAIN, CNTRL                           564
     10 FORMAT (I1, 7X, O4, 2X, O6, O16)                              565
      IF(.NOT.                                                        566
     .   (KEY.EQ.1)                                                   567
     25 .)GO TO 99996                                                 568
      MAINM(NADR+1) = MAIN                                            569
      K = K + 1                                                       570
      GO TO 99995                                                     571
   99996 IF(.NOT.                                                     572
     30 .       (KEY.EQ.2)                                            573
     .)GO TO 99994                                                    574
      CNTMEM(NADR+1) = XOR (MSK1, CNTRL)                              575
      L = L + 1                                                       576
      GO TO 99995                                                     577
     35 99994 IF(.NOT.                                                578
     .       (KEY.EQ.9)                                               579
     .)GO TO 99993                                                    580
      WRITE (7, 20) K,L                                               581
     20 FORMAT (24H0MEMORY LOAD COMPLETED : , I5, 19H WORDS OF MAIN AND ,  582
     40 .          I5, 32H WORDS OF CONTROL MEMORY LOADED.  ,//)      583
      GO TO 99995                                                     584
   99993 CONTINUE                                                     585
      WRITE (7, 30) KEY                                               586
     30 FORMAT (49H0INVALID CONTROL CHARACTER IN MEMORY LOAD CARD =    ,  587
     45 .          I1, 18H.  RUN TERMINATED.  )                       588
      KEY = 9                                                         589
      ERROR = .TRUE.                                                  590
   99995 CONTINUE                                                     591
      GO TO 99998                                                     592
     50 99997 CONTINUE                                                593
      GO TO 99992                                                     594
   99999 IF(.NOT.                                                     595
     .    (NRUN(11).EQ.2)                                            596
     .)GO TO 99991                                                    597
     55 99998 IF(.NOT.                                                598
     .       (KEY.NE.9)                                               599
     .)GO TO 99989                                                    610
```

```
      READ (5, 40) KEY, MADR, MAINL, MAINR, (CNTFLD(I), I = 1,6)
   40 FORMAT (I1,1X,O4,1X,O3,O3,1X,O2,O1,O2,O2,O4,O8)
      IF(.NOT.
     .   (KEY.EQ.1)
     .)GO TO  99988
      MAINM(MADR+1) = OR (SHIFT (MAINL, 8), MAINR)
      K = K + 1
      GO TO 99987
99988 IF(.NOT.
     .      (KEY.EQ.2)
     .)GO TO 99986
      CNTRL = OR (SHIFT (CNTFLD(1), 3), CNTFLD(2) )
      CNTRL = OR (SHIFT (CNTRL, 4), CNTFLD(3) )
      CNTRL = OR (SHIFT (CNTRL, 4), CNTFLD(4) )
      CNTRL = OR (SHIFT (CNTRL, 10), CNTFLD(5) )
      CNTRL = OR (SHIFT (CNTRL, 22), CNTFLD(6) )
      CNTMEM(MADR+1) = XOR (MSK1, CNTRL)
      L = L + 1
      GO TO 99987
99986 IF(.NOT.
     .      (KEY.EQ.9)
     .)GO TO 99985
      WRITE (7, 20) K, L
      GO TO 99987
99985 CONTINUE
      WRITE (7, 30) KEY
      KEY = 9
      ERROR = .TRUE.
99987 CONTINUE
      GO TO 99990
99989 CONTINUE
      GO TO 99992
99991 CONTINUE
      WRITE (7, 50) NRUN(11)
   50 FORMAT (20HOINVALID NRUN(11) =   ,O4, 18H.  RUN TERMINATED.   )
      ERROR = .TRUE.
99992 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE GETCM (IADR, IBITL, IBITN, IBYTE, IFLAG)                    640
      LOGICAL   FINIS  , ERROR                                               641
      COMMON    CNTMEM(1024)     , ICN    , ICOUNT , IDIN   ,               642
     1          IDOUT  , IDSCR   , ITELM  , MAINM(1024)     ,               643
     2          MBUSS  , MEMC    , MEMM   , MICAD(6), MICDA(3),             644
     3          MICQ   , MSCPAD(16)       , MSR1   , MSR2   ,               645
     4          MRUN(2), FINIS   , ERROR  , MINSTR , MOPND  ,               646
     5          MICRO(25),MCNT   , STIME  , ITIME  , IETIM                  647
      COMMON    KAD    , KDA     , IOLOB                                    648
      IF(.NOT.                                                               649
     .  (IADR.LT.0.OR.IADR.GT.1023)                                          650
     .)GO TO  99999                                                          651
      IFLAG = 1                                                              652
      GO TO 99998                                                            653
99999 IF(.NOT.                                                               654
     .     (IBITL.LT.1.OR.IBITL.GT.47)                                       655
     .)GO TO 99997                                                           656
      IFLAG = 2                                                              657
      GO TO 99998                                                            658
99997 IF(.NOT.                                                               659
     .     (IBITN.LT.1.OR.IBITN.GT.48)                                       660
     .)GO TO 99996                                                           661
      IFLAG = 3                                                              662
      GO TO 99998                                                            663
99996 IF(.NOT.                                                               664
     .     ((IBITL + I3ITN).GT.48)                                           665
     .)GO TO 99995                                                           666
      IFLAG = 5                                                              667
      GO TO 99998                                                            668
99995 CONTINUE                                                               669
      IFLAG = 0                                                              670
      IDR = IADR + 1                                                         671
      ISTBIT = 48 - IBITL                                                    672
      IMSK = SHIFT (MASK(IBITN), ISTBIT)                                     673
      IBYTE = AND (CNTMEM(IDR), IMSK)                                        674
      NMBR = 48 -(IBITL + IBITN)                                             675
      IBYTE = SHIFT (IBYTE, -NMBR)                                           676
99998 CONTINUE                                                               677
      IF(.NOT.                                                               678
     .   (IFLAG.NE.0)                                                        679
     .)GO TO  99994                                                          680
      IBYTE = 0                                                              681
99994 CONTINUE                                                               682
      RETURN                                                                 683
      END                                                                    684
```

```
      SUBROUTINE DCODM                                                  685
      LOGICAL   FINIS  ,  ERROR                                         686
      COMMON    CNTMEM(1024)     ,  ICN    ,  ICOUNT ,  IDIN   ,      687
     1          IOOUT  ,  IDSCR  ,  ITELM  ,  MAINM(1824)      ,      688
     2          MBUSS  ,  MEMC   ,  MEMH   ,  MICAD(6), MICDA(3),     689
     3          MICQ   ,  MSCPAD(16)       ,  MSR1   ,  MSR2   ,      690
     4          NRUN(2),  FINIS  ,  ERROR  ,  MINSTR ,  MOPND  ,      691
     5          MICRO(25),MCNT   ,  STIME  ,  ITIME  ,  IETIM         692
      COMMON    KAD    ,  KDA    ,  IOLOB                             693
      DATA IOFF /14008/                                                 694
      MEMC = MSCPAD(1)                                                  695
      MSCPAD(1) = MSCPAD(1) + 1                                         696
      CALL GETMM (MEMC, 0, 0, MIN, IFLAG)                               697
      MINSTR = MIN                                                      698
      IF(.NOT.                                                          699
     .   (IFLAG.EQ.0)                                                   710
     .)GO TO  99999                                                     711
      MEMH = MINSTR + IOFF                                              712
      CALL GETMM (MEMC, 0, 0, MOP, IFLAG)                               713
      MOPND = MOP                                                       714
      IF(.NOT.                                                          715
     .   (IFLAG.EQ.0)                                                   716
     .)GO TO  99998                                                     717
      MEMC = MOPND + IOFF                                               718
      GO TO 99997                                                       719
99998 CONTINUE                                                          710
      WRITE (7, 10) IFLAG                                               711
   10 FORMAT (43H0INVALID CALL TO GETMM FROM DCODM, IFLAG = , I2, 1H. ) 712
      ERROR = .TRUE.                                                    713
99997 CONTINUE                                                          714
      GO TO 99996                                                       715
99999 CONTINUE                                                          716
      WRITE (7, 10) IFLAG                                               717
      ERROR = .TRUE.                                                    718
99996 CONTINUE                                                          719
      RETURN                                                            720
      END                                                               721
```

```
      SUBROUTINE GETMM (IADR, IBITL, IBITN, IBYTE, IFLAG)
      LOGICAL   FINIS  ,  ERROR
      COMMON    CNTMEM(1024)     ,  ICN    ,  ICOUNT ,  IDIN   ,
     1          IOOUT  ,  IDSCR  ,  ITELM  ,  MAINM(1024)      ,
     2          MBUSS  ,  MEMC   ,  MEMM   ,  MICAD(6), MICDA(3),
     3          MICQ   ,  MSCPAD(16)        ,  MSR1   ,  MSR2   ,
     4          NRUN(2J), FINIS  ,  ERROR  ,  MINSTR ,  MOPND  ,
     5          MICRO(25),MCNT   ,  STIME  ,  ITIME  ,  IETIM
      COMMON    KAD    ,  KDA    ,  IOLOB
      IF(.NOT.
     .   (IADR.LT.0.OR.IADR.GT.1023)
     .)GO TO  99999
      IFLAG = 1
      GO TO 99998
99999 IF(.NOT.
     .      (IBITL.LT.0.OR.IBITL.GT.15)
     .)GO TO 99997
      IFLAG = 2
      GO TO 99998
99997 IF(.NOT.
     .      (IBITN.LT.0.OR.IBITN.GT.16)
     .)GO TO 99996
      IFLAG = 3
      GO TO 99998
99996 IF(.NOT.
     .      ((IBITL + IBITN).GT.15)
     .)GO TO 99995
      IFLAG = 5
      GO TO 99998
99995 CONTINUE
      IFLAG = 0
      IDR = IADR + 1
      IBIT = 16 - IBITL
      IMASK = SHIFT (MASK(IBITN), IBIT)
      IBYTE = AND (MAINM(IDR), IMASK)
      NMBR = 16 - (IBITL + IBITN)
      IBYTE = SHIFT (IBYTE, -NMBR)
99998 CONTINUE
      IF(.NOT.
     .   (IFLAG.NE.0)
     .)GO TO  99994
      IBYTE = 0
99994 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE BOLN (IOP)                                                  737
      LOGICAL   FINIS  ,  ERROR                                              738
      COMMON    CNTME(1024)     ,  ICN    ,  ICOUNT ,  IDIN   ,              739
     1          IDOUT  ,  IDSCR  ,  ITELM  ,  MAINM(1024)      ,            770
     2          MBUSS  ,  MEMC   ,  MEMM   ,  MICAD(6), MICDA(3),           771
     3          MICQ   ,  MSCPAD(16)       ,  MSR1   ,  MSR2   ,            772
     4          NRUN(2),  PINIS  ,  ERROR  ,  MINSTR ,  MOPMD  ,            773
     5          MICRO(25),MCNT   ,  STIME  ,  ITIME  ,  IETIM               774
      COMMON    KAD    ,  KDA    ,  IOLOB                                    775
      DATA LMIGH /17777778/                                                  776
      IF (NRUN(3).EQ.2) WRITE (7,  9)                                        777
    9 FORMAT (27H0SUBROUTINE BOLN  ENTERED.  )                               778
      MOD = MICRO(2)                                                         779
      IAR = MICRO(3) + 1                                                     780
      IBR = MICRO(4) + 1                                                     781
      IOLOB = MSCPAD(132)                                                    782
      IF(.NOT.                                                               783
     .   (IOP.EQ.0)                                                          784
     .)GO TO  99999                                                          785
      IF(.NOT.                                                               786
     .   (ICN.EQ.0)                                                          787
     .)GO TO  99998                                                          788
      ITMP = LMIGH                                                           789
      GO TO 99997                                                            790
99998 IF(.NOT.                                                               791
     .      (ICN.EQ.1)                                                       792
     .)GO TO 99996                                                           793
      ITMP = 0                                                               794
      GO TO 99997                                                            795
99996 CONTINUE                                                               796
      ERROR = .TRUE.                                                         797
      WRITE (7, 10) ICN, IOP                                                 798
   10 FORMAT (17H0INVALID CARRY = ,O6,20H IN BOLN, OP CODE = , I2, 1H.)      799
99997 CONTINUE                                                               810
      GO TO 99995                                                            811
99999 IF(.NOT.                                                               812
     .      (IOP.EQ.1)                                                       813
     .)GO TO 99994                                                           814
      ITMP = AND (MSCPAD(IAR), MSCPAD(IBR) )                                 815
      GO TO 99995                                                            816
99994 IF(.NOT.                                                               817
     .      (IOP.EQ.2)                                                       818
     .)GO TO 99993                                                           819
      ITMP = AND (IDIN, MSCPAD(IBR) )                                        820
      GO TO 99995                                                            821
99993 IF(.NOT.                                                               822
     .      (IOP.EQ.3)                                                       823
     .)GO TO 99992                                                           824
      ITMP = OR (MSCPAD(IAR), MSCPAD(IBR) )                                  825
      GO TO 99995                                                            826
99992 IF(.NOT.                                                               827
     .      (IOP.EQ.4)                                                       828
     .)GO TO 99991                                                           829
      ITMP = OR (IDIN, MSCPAD(IBR) )                                         830
      GO TO 99995                                                            831
99991 IF(.NOT.                                                               832
     .      (IOP.EQ.5)                                                       833
```

```
                .)GO TO 99990                                                   824
                 ITMP = XOR (MSCPAD(IAR), MSCPAD(IBR) )                         825
60               GO TO 99995                                                    826
             99990 IF(.NOT.                                                     827
                .      (IOP.EQ.6)                                               828
                .)GO TO 99989                                                   829
                 ITMP = XOR (IDIN, MSCPAD(IBR) )                                830
65           99989 CONTINUE                                                     831
             99995 CONTINUE                                                     832
                 IF(.NOT.                                                       833
                .   (NRUN(3).EQ.2)                                              834
                .)GO TO  99988                                                  835
70               WRITE (7, 25) IOP, ITMP                                        836
                25 FORMAT ( 7H0INSTR , 02, 24H EXECUTED, TEMP VALUE =  ,06,  1H.)  837
             99988 CONTINUE                                                     838
                 IF(.NOT.                                                       839
                .   (MOD.EQ.7)                                                  840
75               .)GO TO  99987                                                 841
                 MICQ = ITMP                                                    842
                 IDOUT = ITMP                                                   843
                 GO TO 99986                                                    844
             99987 CONTINUE                                                     845
80               MSCPAD(IBR) = ITMP                                             846
                 CALL INMOD (IAR, IBR)                                          847
             99986 CONTINUE                                                     848
                 RETURN                                                         849
                 END                                                            850
```

```
SUBROUTINE INVRT      74/74   OPT=1                    FTN 4.2+74335      05/11/75  16.03.49.

                      SUBROUTINE INVRT (IOP)                                                    851
                      LOGICAL   FINIS  ,  ERROR                                                 852
                      COMMON    CNTMEM(1)24)      ,  ICN    ,  ICOUNT ,  IDIN   ,             853
                     1          IDOUT  ,  IDSCR   ,  ITELM  ,  MAINM(1)24)      ,             854
                     2          MBUSS  ,  MEMC    ,  MEMM   ,  MICAD(6), MICDA(3),            855
                     3          MICQ   ,  MSCPAD(16)        ,  MSR1   ,  MSR2   ,             856
                     4          NRUN(2)),  FINIS  ,  ERROR  ,  MINSTR ,  MOPND  ,             857
                     5          MICRO(25),MCNT    ,  STIME  ,  ITIME  ,  IETIM                858
                      COMMON    KAD    ,  KDA     ,  IOLOB                                     859
                      DATA LHIGH /177700/                                                       860
                      IF (NRUN(3).EQ.2) WRITE (7,  5)                                           861
                    5 FORMAT (27H0SUBROUTINE INVRT ENTERED.  )                                  862
                      MOD = MICRO(2)                                                            863
                      IAR = MICRO(3) + 1                                                        864
                      IBR = MICRO(4) + 1                                                        865
                      IOLOB = MSCPAD(IBR)                                                       866
                      IF(.NOT.                                                                  867
                     .   (IOP.EQ.7)                                                             868
                     .)GO TO  99999                                                             869
                      ITMP = COMPL(MSCPAD(IAR) )                                                870
                      GO TO 99998                                                               871
                99999 IF(.NOT.                                                                  872
                     .      (IOP.EQ.8)                                                          873
                     .)GO TO 99997                                                              874
                      ITMP = COMPL (IDIN)                                                       875
                      GO TO 99998                                                               876
                99997 IF(.NOT.                                                                  877
                     .      (IOP.EQ.9)                                                          878
                     .)GO TO 99996                                                              879
                      ITMP = COMPL (MSCPAD(IBR) )                                               880
                      GO TO 99998                                                               881
                99996 IF(.NOT.                                                                  882
                     .      (IOP.EQ.10)                                                         883
                     .)GO TO 99995                                                              884
                      ITMP = COMPL (MIC2)                                                       885
                99995 CONTINUE                                                                  886
                99998 CONTINUE                                                                  887
                      ITMP = AND(ITMP, LHIGH)                                                   888
                      IF(.NOT.                                                                  889
                     .   (ICN.EQ.8)                                                             890
                     .)GO TO  99994                                                             891
                      ITMP = ITMP + LHIGH                                                       892
                      ITMP = AND (ITMP, LHIGH)                                                  893
                      GO TO 99993                                                               894
                99994 IF(.NOT.                                                                  895
                     .      (ICN.LT.0.OR.ICN.GT.1)                                              896
                     .)GO TO 99992                                                              897
                      WRITE (7, 10) ICN, IOP                                                    898
                   10 FORMAT (17H0INVALID CARRY = ,O8,21H IN INVRT, OP CODE = ,I2,1H.)          899
                      ERROR = .TRUE.                                                            910
                99992 CONTINUE                                                                  911
                99993 CONTINUE                                                                  912
                      IF(.NOT.                                                                  913
                     .   (NRUN(3).EQ.2)                                                         914
                     .)GO TO  99991                                                             915
                      WRITE (7, 25) IOP, ITMP                                                   916
                   25 FORMAT ( 7H0INSTR , O2, 24H EXECUTED, TEMP VALUE = ,O6,  1H.)             917
```

```
      99991 CONTINUE                                                    918
            IF(.NOT.                                                    919
60          .   (MOD.EQ.7)                                             910
            .)GO TO  99990                                              911
            MICQ = ITMP                                                 912
            IDOUT = ITMP                                                913
            GO TO 99989                                                 914
65    99990 CONTINUE                                                    915
            MSCPAD(IBR) = ITMP                                          916
            CALL INMOD (IAR, IBR)                                       917
      99989 CONTINUE                                                    918
            RETURN                                                      919
70          END                                                         920
```


85

```
SUBROUTINE CMPLT      74/74   OPT=1                    FTN 4.2+74355      05/11/75  10.03.52.

                      SUBROUTINE CMPLT (IOP)                                                  921
                      LOGICAL    FINIS  ,  ERROR                                              922
                      COMMON     CNTMEM(1024)     ,  ICN    ,  ICOUNT ,  IOIN   ,           923
                     1           IOOUT  ,  IOSCR  ,  ITELM  ,  MAINM(1024)      ,           924
     5               2           MBUSS  ,  MEMC   ,  MEMM   ,  MICAD(6), MICDA(3),          925
                     3           MICQ   ,  MSCPAD(16)       ,  MSR1   ,  MSR2   ,          926
                     4           NRUN(2), FINIS  ,  ERROR   ,  MINSTR ,  MOPND  ,           927
                     5           MICRO(25),MCNT  ,  STIME   ,  ITIME  ,  IETIM              928
                      COMMON     KAD    ,  KDA    ,  IOLOB                                   929
    10                DATA LHIGH /177777B/                                                  930
                      IF (NRUN(3).EQ.2) WRITE (7,  5)                                       931
                    5 FORMAT (27H0SUBROUTINE COMPL ENTERED.  )                              932
                      MOD = MICRO(2)                                                        933
                      IAR = MICRO(3) + 1                                                    934
    15                IBR = MICRO(4) + 1                                                    935
                      IOLOB = MSCPAD(I3R)                                                   936
                      IF(.NOT.                                                              937
                     .   (IOP.EQ.11)                                                        938
                     .)GO TO  99999                                                         939
    20                ITMP = COMPL(MSCPAD(IAR))                                           940
                      GO TO 99998                                                           941
                99999 IF(.NOT.                                                              942
                     .      (IOP.EQ.12)                                                     943
                     .)GO TO 99997                                                          944
    25                ITMP = COMPL (IOIN)                                                 945
                      GO TO 99998                                                           946
                99997 IF(.NOT.                                                              947
                     .      (IOP.EQ.13)                                                     948
                     .)GO TO 99996                                                          949
    30                ITMP = COMPL (MSCPAD(IBR))                                          950
                      GO TO 99998                                                           951
                99996 IF(.NOT.                                                              952
                     .      (IOP.EQ.14)                                                     953
                     .)GO TO 99995                                                          954
    35                ITMP = COMPL (MICQ)                                                 955
                99995 CONTINUE                                                              956
                99998 CONTINUE                                                              957
                      ITMP = AND (ITMP, LHIGH)                                              958
                      ITMP = ITMP + ICN                                                     959
    40                ITMP = AND (ITMP, LHIGH)                                            960
                      IF(.NOT.                                                              961
                     .   (NRUN(3).EQ.2)                                                     962
                     .)GO TO  99994                                                         963
                      WRITE (7, 25) IOP, ITMP                                             964
    45              25 FORMAT ( 7H0INSTR , O2, 24H EXECUTED, TEMP VALUE =  ,O6,  1H.)     965
                99994 CONTINUE                                                              966
                      IF(.NOT.                                                              967
                     .   (MOD.EQ.7)                                                         968
                     .)GO TO  99993                                                         969
    50                MICQ = ITMP                                                         970
                      IOOUT = ITMP                                                          971
                      GO TO 99992                                                           972
                99993 CONTINUE                                                              973
                      MSCPAD(I8R) = ITMP                                                    974
    55                CALL INMOD (IAR, I3R)                                               975
                99992 CONTINUE                                                              976
                      RETURN                                                                977
                      END                                                                   978
```

```
SUBROUTINE TRINC      74/74   OPT=1                    FTN 4.2+74355      05/11/75  10.03.56.

                      SUBROUTINE TRINC (IOP)                                                    979
                      LOGICAL    FINIS  ,  ERROR                                                980
                      COMMON     CNTNEW(1024)      ,  ICN    ,  ICOUNT ,  IOIN   ,            981
                     1           IOOUT  ,  IOSCR   ,  ITELM  ,  MAINM(1024)    ,            982
                     2           MBUSS  ,  MEMC    ,  MEMM   ,  MICAD(6), MICDA(3),           983
                     3           MICQ   ,  MSCPAD(16)        ,  MSR1   ,  MSR2   ,          984
                     4           NRUN(2.),  FINIS  ,  ERROR  ,  MINSTR ,  MOPND  ,          985
                     5           MICRO(23),MCNT    ,  STIME  ,  ITIME  ,  IETIM             986
                      COMMON     KAD    ,  KD4     ,  IOLDB                                   987
                      DATA LHIGH /1777773/                                                    988
                      IF (NRUN(3).EQ.2) WRITE (7,  5)                                         989
                    5 FORMAT (27H0SUBROUTINE TRINC ENTERED.  )                                990
                      MOD = MICRO(2)                                                          991
                      IAR = MICRO(3) + 1                                                      992
                      IBR = MICRO(4) + 1                                                      993
                      IOLDB = MSCPAD(I3R)                                                     994
                      IF(.NOT.                                                                995
                     .    (IOP.EQ.15)                                                         996
                     .)GO TO  99999                                                          997
                      ITMP = MSCPAD(IAR) + ICN                                                998
                      GO TO 99998                                                             999
                  99999 IF(.NOT.                                                             1000
                     .      (IOP.EQ.15)                                                      1001
                     .)GO TO 99997                                                           1002
                      ITMP = IOIN + ICN                                                       1003
                      GO TO 99998                                                             1004
                  99997 IF(.NOT.                                                             1005
                     .      (IOP.EQ.17)                                                      1006
                     .)GO TO 99996                                                           1007
                      ITMP = MSCPAD(IBR) + ICN                                                1008
                      GO TO 99998                                                             1009
                  99996 IF(.NOT.                                                             1010
                     .      (IOP.EQ.19)                                                      1011
                     .)GO TO 99995                                                           1012
                      ITMP = MICQ + ICN                                                       1013
                  99995 CONTINUE                                                              1014
                  99998 CONTINUE                                                              1015
                      ITMP = AND (ITMP, LHIGH)                                                1016
                      IF(.NOT.                                                                1017
                     .   (NRUN(3).EQ.2)                                                       1018
                     .)GO TO  99994                                                          1019
                      WRITE (7, 25) IOP, ITMP                                                 1020
                   25 FORMAT (7HCINSTR ,O2,24H EXECUTED, TEMP VALUE =  ,O8, 1H.)              1021
                  99994 CONTINUE                                                              1022
                      IF(.NOT.                                                                1023
                     .   (MOD.EQ.7)                                                           1024
                     .)GO TO  99993                                                          1025
                      MICQ = ITMP                                                             1026
                      IOOUT = ITMP                                                            1027
                      GO TO 99992                                                             1028
                  99993 CONTINUE                                                              1029
                      MSCPAD(I3R) = ITMP                                                      1030
                      CALL INMOD (IAR, IBR)                                                   1031
                  99992 CONTINUE                                                              1032
                      RETURN                                                                  1033
                      END                                                                     1034
```

```
      SUBROUTINE DECTR (IOP)                                                    1035
      LOGICAL   FINIS  ,  ERROR                                                 1036
      COMMON    CNTMEM(1024)     ,  ICN    ,  ICOUNT ,  IDIN   ,              1037
     1          IDOUT  ,  IDSCR  ,  ITELM  ,  MAINM(1024)      ,              1038
     2          MBUSS  ,  MEMC   ,  MEMM   ,  MICAD(6), MICDA(3),             1039
     3          MICQ   ,  MSCPAD(16)        ,  MSR1   ,  MSR2   ,             1040
     4          NRUN(2), FINIS  ,  ERROR  ,  MINSTR ,  MOPND  ,               1041
     5          MICRO(25),MCNT   ,  STIME  ,  ITIME  ,  IETIM                 1042
      COMMON    KAD    ,  KDA    ,  IOLDB                                     1043
      DATA LMIGH /17777B/                                                       1044
      IF (NRUN(3).EQ.2) WRITE (7, 5)                                            1045
    5 FORMAT (27H0SUBROUTINE DECTR ENTERED.  )                                  1046
      MOD = MICRO(2)                                                            1047
      IAR = MICRO(3) + 1                                                        1048
      IBR = MICRO(4) + 1                                                        1049
      IOLDB = MSCPAD(IBR)                                                       1050
      IF(.NOT.                                                                  1051
     .   (IOP.EQ.19)                                                            1052
     .)GO TO  99999                                                             1053
      ITMP = MSCPAD(IAR)                                                        1054
      GO TO 99998                                                               1055
99999 IF(.NOT.                                                                  1056
     .      (IOP.EQ.20)                                                        1057
     .)GO TO 99997                                                              1058
      ITMP = IDIN                                                               1059
      GO TO 99998                                                               1060
99997 IF(.NOT.                                                                  1061
     .      (IOP.EQ.21)                                                        1062
     .)GO TO 99996                                                              1063
      ITMP = MSCPAD(IBR)                                                        1064
      GO TO 99998                                                               1065
99996 IF(.NOT.                                                                  1066
     .      (IOP.EQ.22)                                                        1067
     .)GO TO 99995                                                              1068
      ITMP = MICQ                                                               1069
99995 CONTINUE                                                                  1070
99998 CONTINUE                                                                  1071
      IF(.NOT.                                                                  1072
     .   (ICN.EQ.0)                                                             1073
     .)GO TO  99994                                                             1074
      ITMP = ITMP + LMIGH                                                       1075
      ITMP = AND (ITMP, LMIGH)                                                  1076
      GO TO 99993                                                               1077
99994 IF(.NOT.                                                                  1078
     .      (ICN.LT.6.OR.ICN.GT.1)                                             1079
     .)GO TO 99992                                                              1080
      WRITE (7, 10) ICN, IOP                                                    1081
   10 FORMAT (17H0INVALID CARRY = ,O8,21H IN DECTR, OP CODE = ,I2,1H.)          1082
99992 CONTINUE                                                                  1083
99993 CONTINUE                                                                  1084
      IF(.NOT.                                                                  1085
     .   (NRUN(3).EQ.2)                                                         1086
     .)GO TO  99991                                                             1087
      WRITE (7, 25) IOP, ITMP                                                   1088
   25 FORMAT ( 7H0INSTR , O2, 24H EXECUTED, TEMP VALUE = ,O6,  1H.)             1089
99991 CONTINUE                                                                  1090
      IF(.NOT.                                                                  1091
```

```
                     .   (MOD.EQ.7)                                          1032
                     .)GO TO  99990                                          1033
60                   MICQ = ITMP                                             1034
                     IDOUT = ITMP                                            1035
                     GO TO 99989                                             1036
                99990 CONTINUE                                               1037
                     MSCPAD(ISR) = ITMP                                      1038
65                   CALL INMOD (IAR, ISR)                                   1039
                99989 CONTINUE                                               1100
                     RETURN                                                  1101
                     END                                                     1102
```

```
SUBROUTINE SUM        74/74   OPT=1                    FTN 4.2+74355      05/11/76  10.04.13.

                      SUBROUTINE SUM (IOP)                                                  1103
                      LOGICAL    FINIS  ,  ERROR                                            1104
                      COMMON     CNTMEM(1024)     ,  ICN    ,  ICOUNT ,  IDIN   ,         1105
                     1           IDOUT  ,  IDSCR  ,  ITELM  ,  MAINM(1024)      ,         1106
                     2           MBUSS  ,  MEMC   ,  MEMM   ,  MICAD(6), MICOA(3),        1107
                     3           MICQ   ,  MSCPAD(16)       ,  MSR1   ,  MSR2   ,         1108
                     4           NRUN(23),  FINIS  ,  ERROR  ,  MINSTR ,  MOPNO  ,         1109
                     5           MICRO(25),MCNT   ,  STIME  ,  ITIME  ,  IETIM            1110
                      COMMON     KAD    ,  KDA    ,  IOLDB                                 1111
                      DATA LHIGH /177778/                                                   1112
                      IF (NRUN(3).EQ.2) WRITE (7, 5)                                        1113
                    5 FORMAT (27H0SUBROUTINE  SUM  ENTERED.  )                              1114
                      MOD = MICRO(2)                                                        1115
                      IAR = MICRO(3) + 1                                                    1116
                      IBR = MICRO(4) + 1                                                    1117
                      IOLDB = MSCPAD(I3R)                                                   1118
                      IF(.NOT.                                                              1119
                     .   (IOP.EQ.23)                                                        1120
                     .)GO TO  99999                                                         1121
                      ITMP = MSCPAD(IAR) + MSCPAD(IBR) + ICN                                1122
                      GO TO 99998                                                           1123
                99999 IF(.NOT.                                                              1124
                     .      (IOP.EQ.24)                                                    1125
                     .)GO TO 99997                                                          1126
                      ITMP = IDIN + MSCPAD(IBR) + ICN                                       1127
                      GO TO 99998                                                           1128
                99997 IF(.NOT.                                                              1129
                     .      (IOP.EQ.25)                                                    1130
                     .)GO TO 99996                                                          1131
                      ITMP = MSCPAD(IAR) + MICQ + ICN                                       1132
                      GO TO 99998                                                           1133
                99996 IF(.NOT.                                                              1134
                     .      (IOP.EQ.26)                                                    1135
                     .)GO TO 99995                                                          1136
                      ITMP = IDIN + MICQ + ICN                                              1137
                99995 CONTINUE                                                              1138
                99998 CONTINUE                                                              1139
                      ITMP = AND (ITMP, LHIGH)                                              1140
                      IF(.NOT.                                                              1141
                     .   (NRUN(3).EQ.2)                                                     1142
                     .)GO TO  99994                                                         1143
                      WRITE (7, 25) IOP, ITMP                                               1144
                   25 FORMAT ( 7H0INSTR , O2, 24H EXECUTED. TEMP VALUE = ,O6,  1H.)         1145
                99994 CONTINUE                                                              1146
                      IF(.NOT.                                                              1147
                     .   (MOD.EQ.7)                                                         1148
                     .)GO TO  99993                                                         1149
                      MICQ = ITMP                                                           1150
                      IDOUT = ITMP                                                          1151
                      GO TO 99992                                                           1152
                99993 CONTINUE                                                              1153
                      MSCPAD(I3R) = ITMP                                                    1154
                      CALL INMOD (IAR, I3R)                                                 1155
                99992 CONTINUE                                                              1156
                      RETURN                                                                1157
                      END                                                                   1158
```

```
      SUBROUTINE DIFF (IOP)                                                  1199
      LOGICAL   FINIS  , ERROR                                               1150
      COMMON    CNTMEM(1024)     , ICN    , ICOUNT , IDIN   ,               1151
     1          IDOUT  , IDSCR  , ITELM  , MAINM(1024)     ,               1152
     2          MBUSS  , MEMC   , MEMM   , MICAD(6), MICDA(3),             1153
     3          MICQ   , MSCPAD(16)      , MSR1   , MSR2   ,               1154
     4          NRUN(2), FINIS  , ERROR  , MINSTR , MOPND  ,               1155
     5          MICRO(25),MCNT  , STIME  , ITIME  , IETIM                  1156
      COMMON    KAD    , KDA    , IOLDB                                    1157
      DATA LHIGH /177770/                                                    1158
      IF (NRUN(3).EQ.2) WRITE (7, 5)                                         1159
    5 FORMAT (27H0SUBROUTINE DIFF  ENTERED.  )                               1170
      MOD = MICRO(2)                                                         1171
      IAR = MICRO(3) + 1                                                     1172
      IBR = MICRO(4) + 1                                                     1173
      IOLDB = MSCPAD(13)                                                     1174
      IF(.NOT.                                                               1175
     .   (IOP.EQ.27)                                                         1176
     .)GO TO  99999                                                          1177
      ITMP = MSCPAD(IAR) + AND(COMPL(MSCPAD(IBR)), LHIGH) + ICN              1178
      GO TO 99998                                                            1179
99999 IF(.NOT.                                                               1180
     .     (IOP.EQ.28)                                                       1181
     .)GO TO 99997                                                           1182
      ITMP = MSCPAD(IBR) + AND(COMPL(MSCPAD(IAR)), LHIGH) + ICN              1183
      GO TO 99998                                                            1184
99997 IF(.NOT.                                                               1185
     .     (IOP.EQ.29)                                                       1186
     .)GO TO 99996                                                           1187
      ITMP = IDIN + AND(COMPL(MSCPAD(IBR)), LHIGH) + ICN                     1188
      GO TO 99998                                                            1189
99996 IF(.NOT.                                                               1190
     .     (IOP.EQ.30)                                                       1191
     .)GO TO 99995                                                           1192
      ITMP = MSCPAD(13) + AND(COMPL(IDIN), LHIGH) + ICN                      1193
      GO TO 99998                                                            1194
99995 IF(.NOT.                                                               1195
     .     (IOP.EQ.31)                                                       1196
     .)GO TO 99994                                                           1197
      ITMP = IDIN + AND(COMPL(MICQ), LHIGH) + ICN                            1198
99994 CONTINUE                                                               1199
99998 CONTINUE                                                               1200
      ITMP = AND (ITMP, LHIGH)                                               1201
      IF(.NOT.                                                               1202
     .   (NRUN(3).EQ.2)                                                      1203
     .)GO TO  99993                                                          1204
      WRITE (7, 25) IOP, ITMP                                                1205
   25 FORMAT ( 7H0INSTR , O2, 24H EXECUTED, TEMP VALUE =  ,O6,  1H.)         1206
99993 CONTINUE                                                               1207
      IF(.NOT.                                                               1208
     .   (MOD.EQ.7)                                                          1209
     .)GO TO  99992                                                          1210
      MICQ = ITMP                                                            1211
      IDOUT = ITMP                                                           1212
      GO TO 99991                                                            1213
99992 CONTINUE                                                               1214
      MSCPAD(IBR) = ITMP                                                     1215
      CALL INMOD(IAR, ISR)                                                   1216
99991 CONTINUE                                                               1217
      RETURN                                                                 1218
      END                                                                    1219
```

```
SUBROUTINE PRBIN      74/74  OPT=1                    FTN 4.2+74335      05/11/75  10.04.19.

                      SUBROUTINE PRBIN                                                  1220
                      LOGICAL   FINIS  ,  ERROR                                         1221
                      COMMON    CNTMEM(1024)     ,  ICN    ,  ICOUNT ,  IDIN   ,      1222
                     1          IDOUT  ,  IDSCR  ,  ITELM  ,  MAINM(1024)      ,      1223
                     2          MBUSS  ,  MEMC   ,  MEMM   ,  MICAD(8), MICDA(3),     1224
                     3          MICQ   ,  MSCPAD(16)       ,  MSR1   ,  MSR2   ,      1225
                     4          MRUN(2), PINIS  ,  ERROR   ,  MINSTR ,  MOPND  ,      1226
                     5          MICRO(25),MCNT  ,  STIME   ,  ITIME  ,  IETIM         1227
                      COMMON    KAD    ,  KDA   ,  IOLOB                               1228
                      IF (MICRO(6).EQ.1) MSR1 = 0                                       1229
                      IF(.NOT.                                                          1230
                     .   (MICRO(10).EQ.1)                                               1231
                     .)GO TO  99999                                                     1232
                      MBUSS = MAINM(MEMC + 1)                                           1233
                      IDIN = MBUSS                                                      1234
                99999 CONTINUE                                                          1235
                      IF(.NOT.                                                          1236
                     .   (MICRO(13).EQ.1)                                               1237
                     .)GO TO  99998                                                     1238
                      MBUSS = NOT (OR (MICAD(KAD), IDSCR) )                             1239
                      IDIN = MBUSS                                                      1240
                99998 CONTINUE                                                          1241
                      IF (MICRO(15).EQ.1) MBUSS = OR (MBUSS, MSR1)                      1242
                      ICN = MICRO(17)                                                   1243
                      IF(.NOT.                                                          1244
                     .   (MICRO(20).EQ.1)                                               1245
                     .)GO TO  99997                                                     1246
                      KDA = 1                                                           1247
                      GO TO 99996                                                       1248
                99997 IF(.NOT.                                                          1249
                     .      (MICRO(21).EQ.1)                                           1250
                     .)GO TO 99995                                                      1251
                      KDA = 2                                                           1252
                      GO TO 99996                                                       1253
                99995 IF(.NOT.                                                          1254
                     .      (MICRO(22).EQ.1)                                           1255
                     .)GO TO 99994                                                      1256
                      KDA = 3                                                           1257
                      GO TO 99996                                                       1258
                99994 CONTINUE                                                          1259
                      KDA = 0                                                           1260
                99996 CONTINUE                                                          1261
                      IF(.NOT.                                                          1262
                     .   (MICRO(23).EQ.1)                                               1263
                     .)GO TO  99993                                                     1264
                      KAD = MICRO(25)                                                   1265
                      IF (KAD.EQ.0) KAD = 3                                             1266
                99993 CONTINUE                                                          1267
                      RETURN                                                            1268
                      END                                                               1269
```

```
      SUBROUTINE BINRY                                                  1270
      LOGICAL   FINIS  , ERROR                                          1271
      COMMON    CNTMEM(1024)     , ICN    , ICOUNT , IDIN   ,          1272
     1          IDOUT  , IDSCR   , ITELM  , MAINM(1024)     ,          1273
     2          MBUSS  , MEMC    , MEMM   , MICAD(6), MICDA(3),        1274
     3          MICQ   , MSCPAD(16)       , MSR1   , MSR2   ,          1275
     4          NRUN(2), PINIS   , ERROR  , MINSTR , MOPND  ,          1276
     5          MICRO(25),MCNT   , STIME  , ITIME  , IETIM             1277
      COMMON    KAD    , KDA     , IOLDB                                1278
   10 DATA MSK10 /17773/                                                1279
      DATA IOFF /14000/                                                 1280
      IF (MICRO(11).EQ.1) MBUSS = IDOUT                                 1281
      IF (MICRO(7).EQ.1) MSR2 = MSR1                                    1282
      IF(.NOT.                                                          1283
   15.   (MICRO(8).EQ.1)                                               1284
     .)GO TO  99999                                                     1285
      MEMM = MICRO(5)                                                   1286
      GO TO 99998                                                       1287
99999 CONTINUE                                                          1288
   20 IF (NRUN(3).EQ.2)  WRITE (7, 4) MEMM                              1289
    4 FORMAT (31H M-REG LOAD BIT NOT SET AT LOC , O4,19H - RUN HALTED. ) 1290
      MEMM = 0                                                          1291
      MICRO(5) = 0                                                      1292
      ERROR = .TRUE.                                                    1293
   25 99998 CONTINUE                                                    1294
      IF(.NOT.                                                          1295
     .   (MICRO(9).EQ.1)                                                1296
     .)GO TO  99997                                                     1297
      M = MEMC + 1                                                      1298
   30 IF(.NOT.                                                          1299
     .   (NRUN(3).EQ.2)                                                 1300
     .)GO TO  99996                                                     1301
      WRITE (7, 5) MEMC, MAINM(M), MBUSS                                1302
    5 FORMAT (17H0MAIN MEMORY LOC , O4, 14H CHANGED FROM , O6, 4H TO ,  1303
   35.       O6, 19H BY CONTROL BIT 35.   )                            1304
99996 CONTINUE                                                          1305
      MAINM(M) = MBUSS                                                  1306
99997 CONTINUE                                                          1307
      IF(.NOT.                                                          1308
   40.   (MICRO(12).EQ.1.AND.MEMM.NE.0)                                 1309
     .)GO TO  99995                                                     1310
      WRITE (7, 10) MEMM                                                1311
   10 FORMAT (34H0ERROR: FETCH BIT ON BUT M-ADDR =  , O6, 1H. )         1312
      ERROR = .TRUE.                                                    1313
   45 99995 CONTINUE                                                    1314
      IF(.NOT.                                                          1315
     .   (MICRO(14).EQ.1)                                              1316
     .)GO TO  99994                                                     1317
      IF (IDOUT.EQ.0) MSR1 = 1                                          1318
   50 99994 CONTINUE                                                    1319
      IF(.NOT.                                                          1320
     .   (MICRO(16).EQ.1)                                              1321
     .)GO TO  99993                                                     1322
      MEMM = OR (MICRO(5), MSR1)                                        1323
   55 IF (NRUN(3).EQ.2) WRITE (7, 30) MEMM                              1324
   30 FORMAT (29H0NEXT 4 MODIFIED BY MSR1 TO:   , O4 )                  1325
99993 CONTINUE                                                          1326
```

```
                IF (MICRO(19).EQ.0) MEMC = AND (MBUSS, MSK10)                    1327
                IF (MICRO(18).EQ.0) ITELM = MBUSS                                1328
60              IF(.NOT.                                                         1329
               .   (KDA.NE.0)                                                    1330
               .)GO TO  99992                                                    1331
                MICDA(KDA) = MBUSS                                               1332
                KDA = 0                                                          1333
65          99992 CONTINUE                                                       1334
                IF (NRUN(3).EQ.2) CALL REPT1                                     1335
                RETURN                                                           1336
                END                                                              1337
```

```
      SUBROUTINE STATS                                                  1338
      LOGICAL   FINIS  ,  ERROR                                         1339
      COMMON    CNTMEM(1024)     ,  ICN    ,  ICOUNT ,  IOIN   ,      1340
     1          IOOUT  ,  IDSCR  ,  ITELM  ,  MAINM(1024)      ,      1341
     2          MBUSS  ,  MEMC   ,  MEMH   ,  MICAD(6), MICDA(3),     1342
     3          MICQ   ,  MSCPAD(16)       ,  MSR1   ,  MSR2   ,      1343
     4          NRUN(2),  FINIS  ,  ERROR  ,  MINSTR ,  MOPND  ,      1344
     5          MICRO(25),MCNT   ,  STIME  ,  ITIME  ,  IETIM         1345
      COMMON    KAD    ,  KDA    ,  IOLOB                             1346
      WRITE (7, 10) STIME, ICOUNT, MCNT, MSCPAD, IOIN, IOOUT, MBUSS,   1347
     .              MEMC, MEMH, MICQ, MICAD, MICDA, ICN,IDSCR,MSR1,MSR2 1348
   10 FORMAT (26H1RUN TERMINATED AT TIME :  ,F15.8,15H SECONDS AFTER   , 1349
     .        I6,11H MICROS OR , I5,17H MACROS EXECUTED.    //         1350
     .        21HOSCRATCHPAD CONTAINS:,/1H , 16( /5X, O8), //          1351
     .        11HODATA IN = , O8,13H, DATA OUT = , O8, 9H, BUSS = , O8, 1352
     .        10H, C-REG = ,O4,10H, H-REG = ,O4,10H, Q-REG = , O6, //  1353
     .        9HOA/D S = ,5(2X,O8),10H, D/A S = ,3(2X,O8),//           1354
     .        11HO CARRY =   ,I2, 20H, INPUT DISCRETES =  , O8, //     1355
     .        9HO MSR1 = , I1, 9H, MSR2 =  , I1 )                      1356
      RETURN                                                            1357
      END                                                               1358
```

```
      SUBROUTINE REPT1                                                  1349
      LOGICAL   FINIS  , ERROR                                          1350
      COMMON    CNTMEM(1024)     , ICN    , ICOUNT , IDIN   ,          1351
     1          IDOUT  , IDSCR   , ITELM  , MAINM(1024)     ,          1352
    9 2          MBUSS  , MENC    , MEMM   , MICRO(6), MICDA(2),        1353
     3          MICQ   , MSCPAD(16)       , MSR1   , MSR2   ,          1354
     4          NRUN(2), PINIS   , ERROR  , MINSTR , MOPND  ,          1355
     5          MICRO(25),MCNT   , STIME  , ITIME  , IETIM             1356
      COMMON    KAD    , KOP     , IOLOB                                1357
   10 INSTR = MICRO(1)                                                  1358
      MOD = MICRO(2)                                                    1359
      IA = MICRO(3) + 1.                                                1370
      IB = MICRO(4) + 1.                                                1371
      I = IA - 1                                                        1372
      J = IB - 1                                                        1373
   19 WRITE (7,10)INSTR,MOD,I,MSCPAD(IA),J,IOLOB,MSCPAD(IB),MICQ2,MICQ  1374
     1          , ITEL2, ITELM                                         1379
   10 FORMAT (18H0EXECUTED OP CODE ,O2,15H WITH MODIFIER ,I1,1H. //     1376
     .        7H A-REG ,O2,12H CONTENTS =  ,O6,// 7H B-REG , O2,       1377
   20 .        22H CONTENTS INITIALLY =  ,O6, 8H, NOW =  ,O6, //        1378
     .        28H Q-REG CONTENTS INITIALLY =  ,O6,8H, NOW =  ,O6, //   1379
     .        28H TELEM CONTENTS INITIALLY =  ,O6,8H, NOW =  ,O6 // )  1380
      ITEL2 = ITELM                                                     1381
      MICQ2 = MICQ                                                      1382
   25 RETURN                                                            1383
      END                                                               1384
      SUBROUTINE REPT2                                                  1385
      RETURN                                                            1386
      END                                                               1387
```

```
*********LQ23      IJB6G4P  //// END OF LIST ////
*********LQ23      IJB6G4P  //// END OF LIST ////
```

DISTRIBUTION

	No. of Copies
Defense Documentation Center Cameron Station Alexandria, Virginia 22314	6 6
Commander US Army Materiel Development and Readiness Command	
ATTN: DRCRD	1
DRCDL	1
5001 Eisenhower Avenue Alexandria, Virginia 22333	
DRSMI-FR, Mr. Strickland	1
-LP, Mr. Voigt	1
-R, Dr. McDaniel	1
Dr. Kobler	1
-RG, Mr. Huff	1
-RGG,	10
-RGC	1
-RGL	1
-RGP	1
-RGT	1
-RGN	1
-RBD	3
-RPR (Record Copy)	1
(Reference Copy)	1