SDS

SCIENTIFIC DATA SYSTEMS

Reference Manual

SDS 92 Computer

SDS 92 BASIC INSTRUCTIONS (CENTRAL PROCESSOR)

Mnemonic	Octal Code	Name
LOAD/STORE		
LDA A,T	64	Load A
LDB A,T	24	Load B
STA A,T	44	Store A
STB A,T	04	Store B
XMA A,T	74	Exchange Memory and A
XMB A,T	34	Exchange Memory and B
FLAG		
XMF A,T	17	Exchange Memory and F
LDF A,T	57	Load F
SFT	0044	Set Flag True
SFF	0042	Set Flag False
INF	0046	Invert Flag
ARITHMETIC		
ADA A,T	62	Add to A
ADB A,T	22	Add to B
ACA A,T	63	Add with Carry to A
ACB A,T	23	Add with Carry to B
SUA A,T	60	Subtract to A
SUB A,T	20	Subtract to B
SCA A,T	61	Subtract with Carry to A
SCB A,T	21	Subtract with Carry to B
MPA A,T	76	Memory Plus A to Memory
MPB A,T	36	Memory Plus B to Memory
MPO A,T	16	Memory Plus One to Memory
MPF A,T	56	Memory Plus Flag to Memory
MUA A,T	13	Multiply A (Optional)
MUB A,T	53	Multiply B (Optional)
DVA A,T	52	Divide AB (Optional)
DVB A,T	12	Divide BA (Optional)
LOGICAL		
ANA A,T	65	AND to A
ANB A,T	25	AND to B
ORA A,T	67	OR to A
ORB A,T	27	OR to B
EOA A,T	66	Exclusive OR to A
EOB A,T	26	Exclusive OR to B
MAA A,T	75	Memory AND A to Memory
MAB A,T	35	Memory AND B to Memory
COMPARISON		
COA A,T	45	Compare Ones with A
COB A,T	05	Compare Ones with B
CMA A,T	47	Compare Magnitude of Memory with A
CMB A,T	07	Compare Magnitude of Memory with B
CEA A,T	46	Compare Memory Equal to A
CEB A,T	06	Compare Memory Equal to B

Mnemonic	Octal Code	Name
BRANCH		
BRU A,T	73	Branch Unconditionally
BRC A,T	32	Branch, Clear Interrupt, and Load Flag
BRL A,T	33	Branch and Load Flag
BFF A,T	31	Branch on Flag False
BFT A,T	71	Branch on Flag True
BDA A,T	70	Branch on Decrementing A
BAX A,T	30	Branch and Exchange A and B
BRM A,T	77	Branch and Mark Place
BMC A,T	37	Branch, Mark Place, and Clear Flag
SHIFT		
CYA A,T	42	Cycle A
CYB A,T	02	Cycle B
CFA A,T	43	Cycle Flag and A
CFB A,T	03	Cycle Flag and B
CYD A,T	02 or 42	Cycle Double
CFD A,T	43	Cycle Flag and Double
CFI A,T	03	Cycle Flag and Double Inverse
CONTROL		
EXU A,T	72	Execute
HLT	0041 or 00000000	Halt
TRAP		
SCT	0061	Set Program-controlled Trap
RCT	0060	Reset Program-controlled Trap
TCT	0160	Test Program-controlled Trap
BREAKPOINT TESTS		
BPT 1	0144	Breakpoint No. 1 Test
BPT 2	0145	Breakpoint No. 2 Test
BPT 3	0146	Breakpoint No. 3 Test
BPT 4	0147	Breakpoint No. 4 Test
INTERRUPTS		
EIR	0051	Enable Interrupt
DIR	0050	Disable Interrupt
IET	0150	Interrupt Enabled Test
AIR	00020001	Arm Interrupts

Legend:

A = address; *A = indirect address; =A = immediate address; T = index tag

SDS 92 COMPUTER
REFERENCE MANUAL

June 1965

SDS

SCIENTIFIC DATA SYSTEMS/1649 Seventeenth Street/Santa Monica, California

Printed in U.S.A.

REVISIONS

This publication, 90 05 05C, supersedes the SDS 92 Computer Reference Manual, 900505B.

Revisions, corrections, and clarifications to the previous edition are indicated by a vertical line in the left or right margin of the page.

CONTENTS

APPENDIXES

FIGURES

TABLES

SDS 92 Computer

I. GENERAL DESCRIPTION

INTRODUCTION

The SDS 92 is a small, high-speed, very low-cost, general-purpose computer designed especially to include application in the following areas:

- High-speed computer-based systems featuring speed and flexibility
- Format conversion featuring complete versatility in formats and equipment involved
- Small, general-purpose applications featuring repetitive, high-speed computation

The SDS 92 has the following characteristics:

- The first commercial computer using monolithic integrated circuits; all flip-flops are integrated
- 12-bit word plus parity bit
- 1.75 μsec memory cycle
- Binary, integer arithmetic
- 12- and 24-bit instructions
- Immediate, direct and indirect addressing
- 2048-word basic memory
- Memory expandable to 4096, 8192, 16,384 or 32,768 words, all directly addressable
- Two independent arithmetic registers, either of which may be an accumulator
- True Index Register; adds no time to execution
- Instruction set comparable to medium-scale computer; includes shift instructions
- 4096 single-bit control outputs; 4096 single-bit sense inputs
- Independent I/O buffer with automatic assembly/disassembly of 6-bit characters to/from words (standard)
- I/O block transfer standard
- Three standard I/O modes:

  6-bit characters, 286,000 characters/second, parity checked and packed 1 per word

  6-bit characters, 286,000 characters/second, parity checked and packed 2 per word

  12-bit parallel words, 572,000 words/second, parity checked
- I/O transfer of 12-bit characters, 286,000 characters/second, parity checked, optional
- Four console sense switches
- Optional features

  Interlace feature for standard I/O buffer

  High-speed multiply and divide instructions, 5 and 13 cycles, respectively

  Many-channel Data Multiplexing System using a second, independent path to memory

  24-bit parallel I/O

  Up to 256 levels of true priority interrupt

  Automatic power fail-safe

  Memory parity checking

  Real-time clock
- Peripheral equipment for the SDS 92

  Keyboard/printer (teletype) with or without paper tape reader and punch, 10 cps

  Paper tape reader and punch, 300 and 60 cps, respectively

  MAGPAK Magnetic Tape System

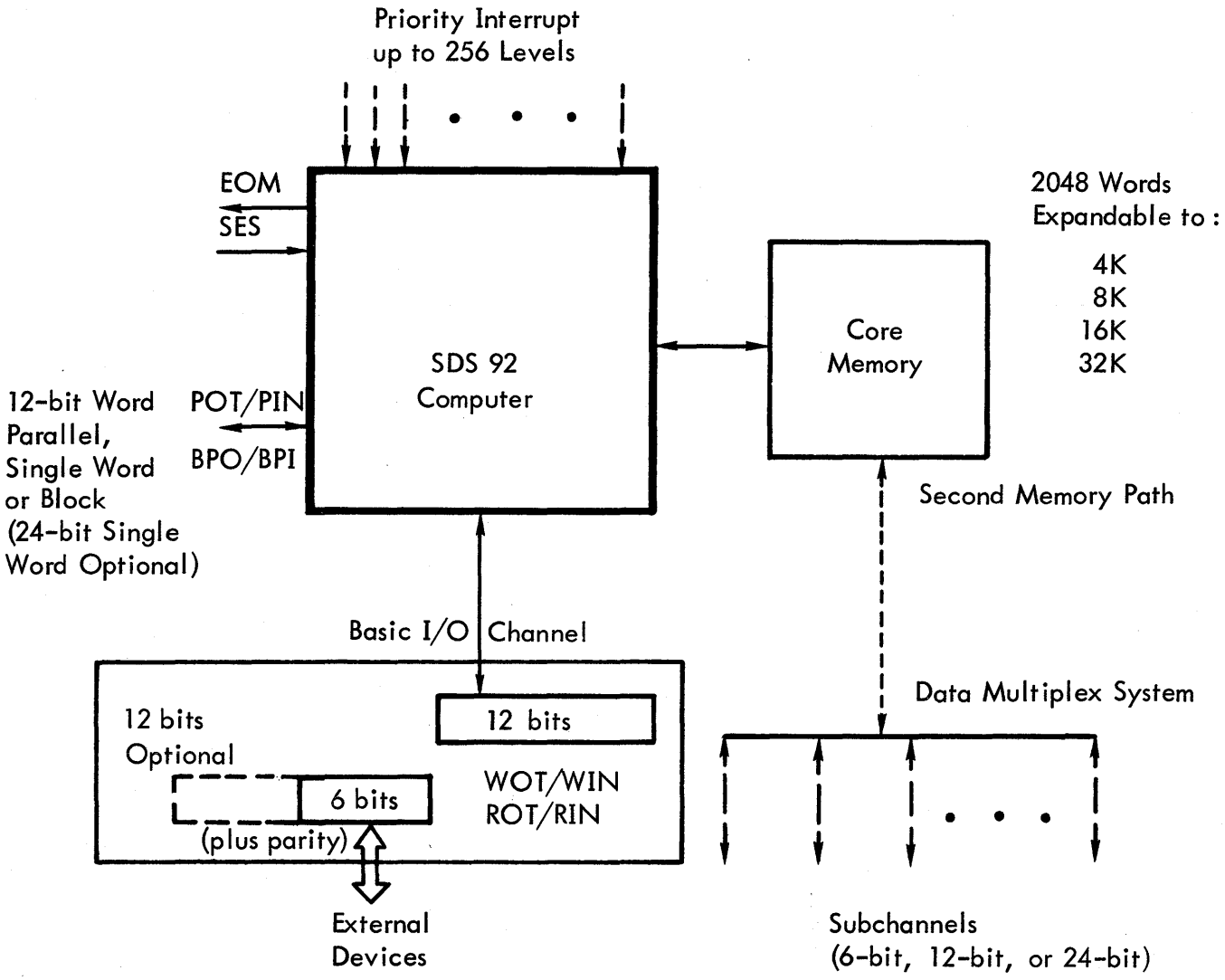
  All equipment in SDS standard peripheral line
- Software

  Basic utility package

  Symbolic assembler

  I/O packages for optional peripheral equipment

  Mathematical subroutines, including floating-point arithmetic, fixed-point multiply and divide, and elementary mathematical functions
- All silicon semiconductors
- Operating temperature range: 10° to 40°C
- Dimensions: 65 in. x 30 in. x 25-1/2 in.
- Power: 1 kva



Items with dotted lines (---) are optional .

Figure 1-1. SDS 92 Computer Configuration

SDS 92 REGISTERS

The 92 Central Processor contains the following arithmetic and control registers. They are full-word, 12-bit registers except as noted.

AVAILABLE TO THE PROGRAMMER (dark lines)

The A Register and the B Register are two independent, 12-bit accumulators. The A Register is also the index register.

The P Register is a 15-bit register that contains the memory address of the current instruction. Unless modified by the program, the contents of P increase by one during one-word instructions and increase by two during two-word instructions.

The Flag Bit Register is a one-bit register used for arithmetic carry, storage and testing.

NOT AVAILABLE TO THE PROGRAMMER (light lines)

The S Register is a 15-bit register that contains the address of the memory location being accessed for instructions or data.

The C Register is a 12-bit register that is used in arithmetic and control operations.

The O Register is a 6-bit register that contains the operation code of the instruction being executed.

The M Register is a 13-bit register that holds each word as it comes from memory. Recopying of a word into memory takes place from the M Register.

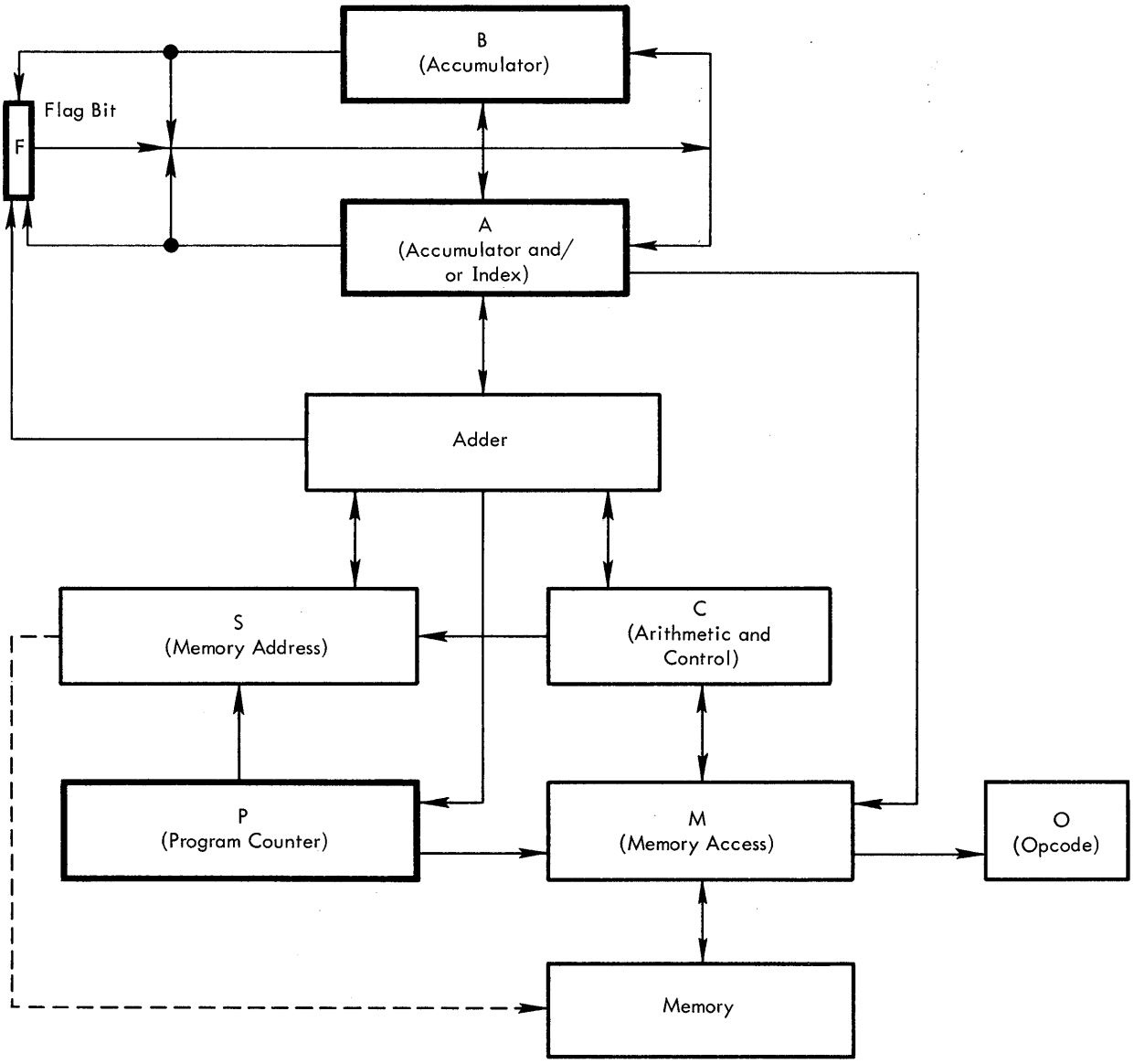


Figure 1-2. Basic Register Flow Diagram

SDS 92 MEMORY

The basic 92 memory contains 2048 words, consisting of 12 bits plus parity. Memory is available in 4096-, 8192-, 16,384- and 32,768-word sizes. The central processor can directly address all memory. Addresses for memory words extend from locations 00000 to 77777 (octal). If the address of the next instruction to be executed is outside of available memory (for less than 32,768-word memories), the 92 executes a halt instruction (00000000); the P Register will contain the requested address + 2. For example, assume a 3777-word memory and

1. the instruction "Branch-to" 4000, or
2. the instruction "Load A" whose location is 3777.

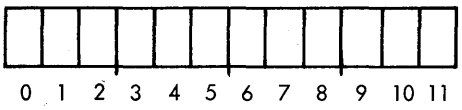
In both cases, a halt occurs after execution of the instruction and the P Register contains 4002 which is outside of available memory. Note that the P Register responds as if there were always 32,768 words of memory. With a 32,768-word memory, the memory is a "wrap-around" or circular memory where the next location after 77777 is 00000. An attempt to read from a location whose address is not available causes zeros to be read. An attempt to store into such a location essentially results in a "no-op" operation, with the next instruction in sequence being executed. A program can use this property to determine the core size available in the machine.

A power fail-safe option is available such that: before accessing each memory word, the computer checks the power to ensure that it can complete the entire read/write cycle. If it detects a power loss, the computer halts.

With the memory parity option, the computer automatically generates even parity or checks for it during each read/write cycle (optional). Setting a control panel parity switch causes the computer to halt automatically in case of parity error detection.

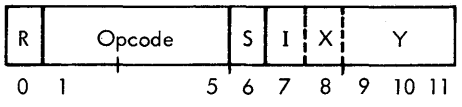
MEMORY WORD FORMATS

A computer word is 12 binary digits (bits) long:



The word format numbers the bits from the left, or most significant end of the word, to the right, or least significant end of the word. This numbering format serves as a basis of reference to bit positions or bit numbers. Octal notation most easily describes the contents of the 12 bits of a word. Thus, one octal digit, 0 through 7, represents three binary digits. For example, the octal number, 0123, represents its binary equivalent, 000 001 010 011.

The 92 instruction word:



offers great versatility both in addressing and indexing capabilities. In most cases, the instruction can select either the A or B for use as the accumulator.

Bit Position	Meaning
0	Register Selection: 1 = A Register † 0 = B Register
1-5	Instruction Code
6	Scratch Pad Bit
7	Indirect Address Bit
8	Index Register Bit
9-11	Part of the Address Field

The flexibility of addressing in the 92 allows bits 7, 8, and 9 through 11 to be used in more than one way as explained below.

ADDRESSING FACILITIES

The 92 has one-word or two-word instructions with the length depending on the addressing mode being used. The addressing modes are:

Type	Addressing Area
Direct, Two Words	Full Memory
Indirect, One Word	Indirect Through Scratch Pad
Indirect, Two Words	Indirect Through Full Memory
Direct, One Word	Scratch Pad
Immediate, One Word	Next Location

Indirect addressing can be cascaded indefinitely.

The standard assembler forms for instructions are:

	Form	Type
Label	Opcode n	Direct, One and Two Words
Label	Opcode n, 1	Direct, Two Words, Index
Label	Opcode *n	Indirect, One and Two Words
Label	Opcode =c	Immediate

where n is a label or a 1- to 5-digit number,
c is a 1- to 4-digit number (<4096).

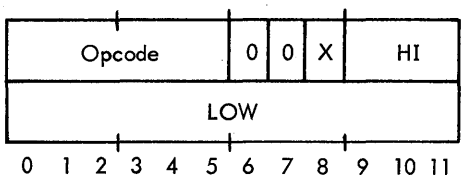
The following section describes the bit configurations that the octal programmer or the symbolic assembler must provide to select the various addressing modes.

Direct Addressing

The programmer selects direct addressing by setting both bits 6 and 7 to zero. Bits 9 through 11 of the first word combine with the entire second word to form a 15-bit address to directly address up to 32,768 words. Bit 8 of the first word can specify indexing.

† The multiply instructions are exceptions.

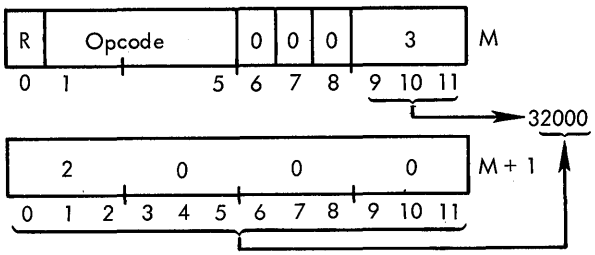
Instruction Form:



Direct Addressing With No Indexing

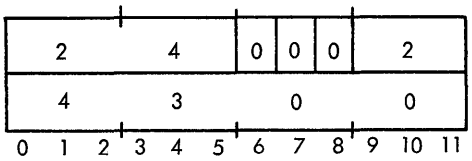
The computer constructs the 15-bit direct address from bits 9, 10, 11 of the instruction word and bits 0 through 11 of the next location.

Example 1:



Assembler Form: LDB 024300

Machine Language Form:



Note: In the standard assembler, a 0 precedes any octal number; nothing precedes a decimal number written as a literal (i.e., immediate).

Direct Addressing With Indexing

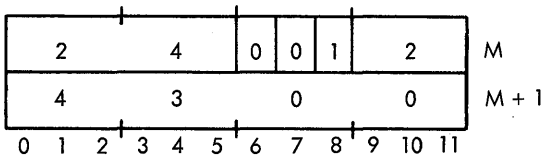
When the programmer sets the Index Bit (number 8) to 1, the computer subtracts the contents of the A Register from the direct address to obtain the effective address.

```
  M M M M M
-   N N N N
-----------
Effective Address
```

Example 2: Assume (A)=0001

Assembler Form: LDB 024300, 1

Machine Language Form:



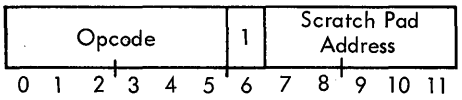
This instruction loads the contents of location 24277 into the B Register.

Note: Although the A Register is only 12 bits long, the top three bits of the address will be modified by indexing if a "borrow" occurs.

Scratch Pad Addressing

Memory locations 00001 to 00037_8 are referred to as the "scratch pad" memory and are special only in that they can be addressed more simply.

When the programmer sets the Scratch Pad Indicator (bit 6) to 1, the instruction addresses one of the 31 scratch pad locations. This allows a complete instruction in a single word. The form of the instruction is:



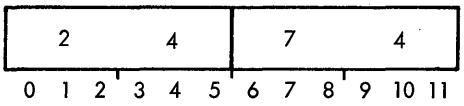
The value of n must be in the range $1 \leqslant n \leqslant 31_{10}$.

Example: Assume

Location	Contents
34	2333

Then the instruction:

LDB 034

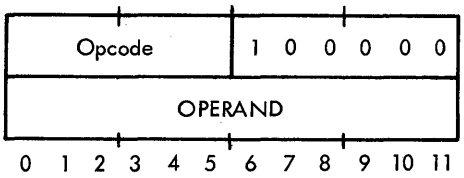


yields (B) = 2333

Immediate Addressing

When the programmer sets the Scratch Pad Indicator to 1 and sets bits 7 through 11 to 00, the instruction acquires its address by adding one to the current contents of the S Register. That is, the next location contains the operand. The computer automatically increments the Program Counter by an additional one to take the next instruction from the word following the immediate operand word.

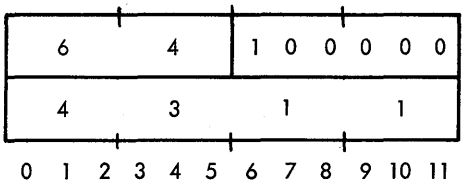
The form of the instruction is:



The standard assembler form is: LDA =, where "=" means "take the following number literally".

Example:

LDA = 04311



In the standard assembler, a 0 precedes any octal number; nothing precedes a decimal number written as a literal (i.e., immediate).

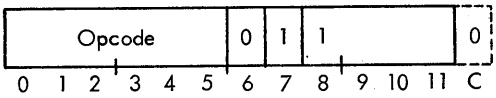
Indirect Addressing

The indirect addressing facility provides two ways of specifying the "pointing" address.

Indirect From Scratch Pad

First, the single-word instruction specifies that the pointing address is within the second 16 locations of the Scratch Pad area (locations 20_8 through 37_8). The specified pointing address used in the standard assembler form must be in the range: $20 \leq n \leq 36$, n even.

In this mode, the instruction bits have the form:



where bit 6 is 0 and bits 7 and 8 are 1.

Bits 8, 9, 10, 11, and C form a five-bit address that selects the first of two contiguous address words. The computer always supplies a least significant fifth bit (C) that is zero, making all such Scratch Pad addresses even. When the addressing logic selects the even address, it reinitiates addressing and interprets that location and the next one as an instruction word-pair without an instruction code. These two (or one) words can specify all forms of addressing.

Bit 8 of the instruction does not effect indexing in this operating mode.

Example:

Location	Contents
34	0002
35	0010
20010	0004

Then,

LDB *034

yields reinitialized addressing on locations 34, 35.

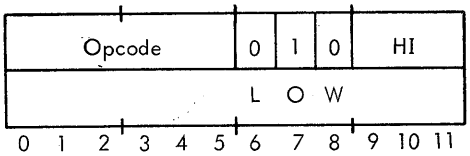
34	0002
35	0010

Since the 0 in bits 6,7,8 of location 34 indicates direct addressing, the instruction does the following:

(20010) = 0004 → B

Indirect Full Addressing

The second form of indirect addressing is:

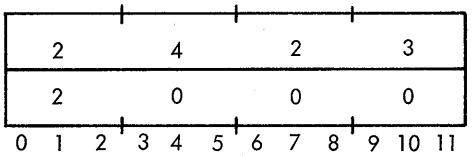


where the 0 in bit 8 specifies that the pointing address consists of a 15-bit address constructed as in Direct Addressing.

Example: Assume

Location	Contents
32000	0001
32001	4020
14020	2200

Take LDB *032000 which is :



This yields:

(32000) = 0001
(32001) = 4020

The logic reinitializes addressing and

(14020) = 2200 → B

The two Pointer words can specify all forms of addressing.

TRAPPING FACILITY

Program-Controlled

The program-controlled trapping facility permits the calling of subroutines with a single instruction of the same form as built-in, machine instructions. The trapping is controlled by the status of the Program-Controlled Trap (PCT) bit. When PCT is a 0, the computer decodes the eight trapping Opcodes as special instructions and executes the Opcode in the unique trapping location determined by the Opcode. When PCT is a 1, the computer decodes the eight Opcodes as normal instructions. The Opcodes, their normal names, and their respective trapping locations follow.

Opcode	Trapping Location	Normal Mnemonic
10	130	POT
50	132	BPO
11	134	WOT
51	136	ROT
14	140	PIN
54	142	BPI
15	144	WIN
55	146	RIN

One of the "branch and mark place" instructions BRM or BMC placed in the trapping location (and location plus one) by the programmer branches to the associated subroutine. The mark information provides the subroutine linkage back to the main program, i.e., the address stored in the mark is the address of the instruction that caused the trap. The program sets, resets, and tests the PCT bit via the instruction:

SET PROGRAM-CONTROLLED TRAP (SCT)

RESET PROGRAM-CONTROLLED TRAP (RCT)

TEST PROGRAM-CONTROLLED TRAP (TCT)

Multiply and Divide

When the high-speed multiply/divide option is not present in a system, an attempt to execute the associated instruction code causes a transfer to a special trap location. These locations are:

Mnemonic	Opcode	Location
MUA	13	124
MUB	53	126
DVA	52	122
DVB	12	120

When the option is absent, the trapping process is always active and cannot be inhibited by the programmer.

Trapping Notes

When a trap occurs, the P counter is not incremented. It is therefore mandatory that only branch instructions be placed in the trap instructions; otherwise, the program goes into an unrecoverable loop:

Assume there is no multiply option.

```
 124  LDA  1000
       :
1200  MUA  5000
```

If MUA is executed, the execution sequence is:

```
1200  MUA
 124  LDA
1200  MUA
 124  LDA
  :    :
  :    :
```

Appendix B-5 contains a complete trap-subroutine example that is useful as a guide to writing subroutine return code.

Nomenclature

Throughout the following discussions, the term "next location" refers to the location immediately following the location of the instruction under discussion. The similar term "next address" is also used.

The term "effective memory location" describes the location in memory from which the computer takes the final operand at the conclusion of all indirect addressing and indexing. The effective memory location is the location whose address is the effective address.

II. MACHINE INSTRUCTIONS

This section describes SDS 92 instructions; they are presented in functional groups. Lists of instructions in functional, numerical and alphabetical order are in Appendices B-7 through B-17.

The following statements apply to the instruction descriptions.

All instruction times are in memory cycles, where each cycle is 1.75 microseconds.

All timings assume that the instruction addresses operands in scratch-pad memory (even though the instruction may, in fact, preclude this mode of addressing). For more comprehensive addressing, add cycles to the given execution times as follows:

Addressing	Cycles
Immediate	0
Direct Full	1
Direct Full with Indexing	1
Indirect Addressing One-Word, even scratch-pad address	P
Indirect Addressing Two-Word, Full address	1 + P

where P is the number of cycles required to process the indirect address pair.

Parentheses denote "contents of", as, for example, (A) represents the contents of the A Register.

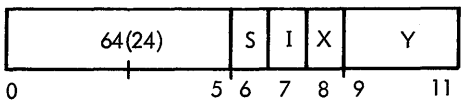
The interrupt system (optional) can interrupt the program sequence at the end of any instruction except as noted.

Those instructions that apply to the A and the B Registers appear with the B Register operation code and mnemonic in parentheses.

With each instruction description is a diagram depicting the instruction format. Preceding this diagram is the mnemonic code and the instruction name. In the diagram, S stands for Scratch Pad Bit, I stands for Indirect Address Bit, X stands for Index Bit, and Y stands for part of the address. The letter M depicts a general memory location.

LOAD/STORE INSTRUCTIONS

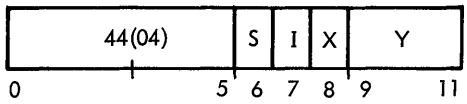
LDA (LDB) LOAD A (LOAD B)



LDA loads the contents of the effective memory location into the A Register.

Registers Affected: A(B) Timing: 2

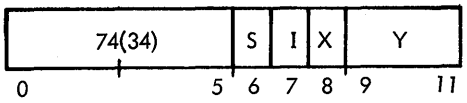
STA (STB) STORE A (STORE B)



STA stores the contents of the A Register in the effective memory location.

Registers Affected: M Timing: 2

XMA (XMB) EXCHANGE M AND A (EXCHANGE M AND B)

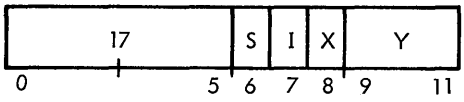


XMA loads the contents of the effective memory location into the A Register and stores the contents of the A Register in the effective memory location.

Registers Affected: A(B), M Timing: 3

FLAG INSTRUCTIONS

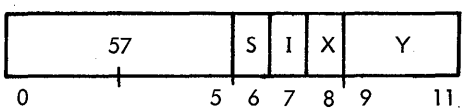
XMF EXCHANGE M AND FLAG



XMF loads the content of the Flag Bit into bit 0 of the effective memory location and loads the content of bit 0 into the Flag Bit; it leaves bit positions 1 through 11 of the effective memory location the same as they were.

Registers Affected: F, M Timing: 3

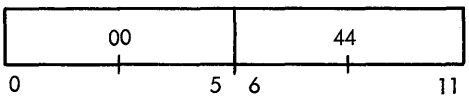
LDF LOAD FLAG



LDF loads the content of bit 0 of the effective memory location into the Flag Bit.

Registers Affected: F Timing: 3

SFT SET FLAG TRUE

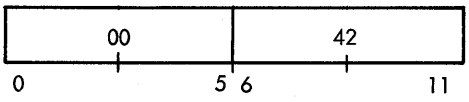


SFT unconditionally sets the Flag Bit to a one.

SFT cannot be interrupted.

Registers Affected: F Timing: 3, 4

SFF SET FLAG FALSE

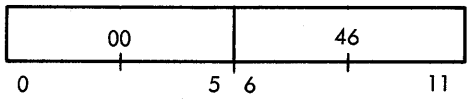


SFF unconditionally resets the Flag Bit to a zero.

SFF cannot be interrupted

Registers Affected: F Timing: 3, 4

INF INVERT FLAG



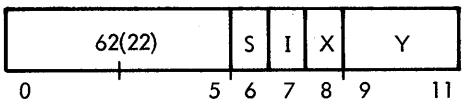
INF unconditionally inverts the Flag Bit. If it is a 1, INF sets it to a 0; if it is a 0, INF sets it to a 1.

INF cannot be interrupted.

Registers Affected: F Timing: 3, 4

ARITHMETIC INSTRUCTIONS

ADA (ADB) ADD TO A (ADD TO B)



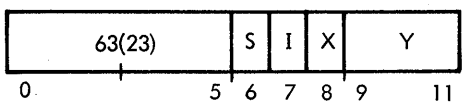
ADA adds the contents of the effective memory location to the contents of A and stores the result in A; it stores the carry from bit 0 of the addition in the Flag Bit.

Registers Affected: A(B), F Timing: 2

Example: Assume
(A) = 4300
(1000) = 3700
(F) = 0 , Flag Bit

Performing ADA 01000 yields
(A) = 0200
(F) = 1

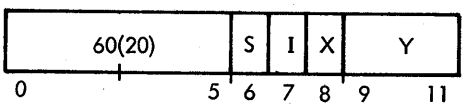
ACA (ACB) ADD WITH CARRY TO A (ADD WITH CARRY TO B)



ACA adds the contents of the effective memory location to the contents of A; it also adds the content of the Flag Bit to bit position 11 of the result. ACA places the final result in A and records the carry from bit 0 in the Flag Bit.

Registers Affected: A(B), F Timing: 2

SUA (SUB) SUBTRACT TO A (SUBTRACT TO B)



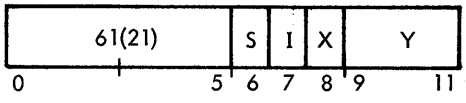
SUA subtracts the contents of the effective memory location from the contents of A and places the result in A; it stores the carry from bit 0 in the Flag (F) bit. [(M) > (A) sets F; (A) ≥ (M) resets F.]

Registers Affected: A(B), F Timing: 2

Example: Assume
(A) = 3003
(10) = 4010
(F) = 0

Performing SUB 010 yields
(A) = 6773
(F) = 1

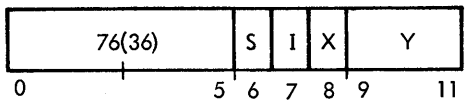
SCA (SCB) SUBTRACT WITH CARRY TO A (SUBTRACT WITH CARRY TO B)



SCA subtracts the content of the effective memory location from the contents of the A Register, then subtracts the content of F from the least significant end of the difference and places the result in A. It places the carry from bit 0 in the Flag (F) bit. [(M)+F > (A) sets F; (A)≥(M)+F resets F.]

Registers Affected: A(B), F Timing: 2

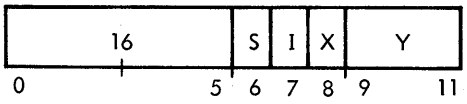
MPA (MPB) MEMORY PLUS A TO MEMORY (MEMORY PLUS B TO MEMORY)



MPA adds the contents of the effective memory location to the contents of A and places the result in the effective memory location; it stores the carry from bit 0 in the Flag Bit.

Registers Affected: M, F Timing: 3

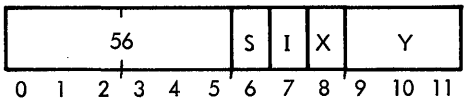
MPO MEMORY PLUS ONE TO MEMORY



MPO increments the contents of the effective memory location by one and places the result back into the same location; it places the carry bit from bit 0 in the Flag Bit.

Registers Affected: M, F Timing: 3

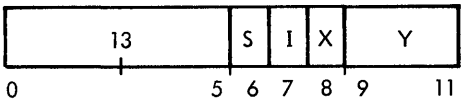
MPF MEMORY PLUS FLAG TO MEMORY



MPF adds the content of the Flag Bit to the contents of the effective memory location at bit position 11 and places the result back into the effective location. The carry from bit 0 of the addition goes into the Flag Bit.

Registers Affected: M, F Timing: 3

MUA MULTIPLY A (Optional)



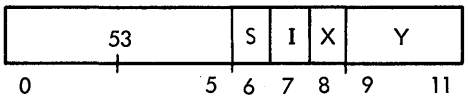
MUA multiplies the contents of A by the contents of the effective memory location and places the product in A and B with the more significant portion in A.

Registers Affected: A, B Timing: 5

Example: Assume
(A) = 3411
(1000) = 0220

Performing MUA 01000 yields
(A) = 0077
(B) = 2420

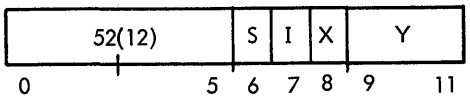
MUB MULTIPLY B (Optional)



MUB multiplies the contents of B by the contents of the effective memory location and places the product in A and B with the more significant portion in A.

Registers Affected: A, B Timing: 5

DVA(DVB) DIVIDE AB (DIVIDE BA) (Optional)



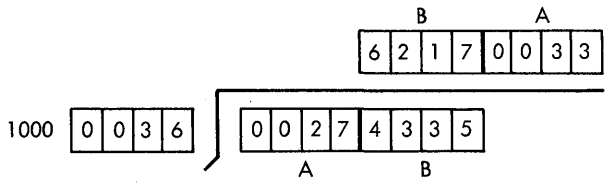
DVA(DVB) divides the contents of the A and B Registers (B and A Registers), treated as a double-precision number, by the contents of the effective memory location and places the quotient in the B Register, with the remainder in the A Register. The A Register (B Register) must initially contain the more significant half of the dividend. The contents of the effective memory location must be greater than the contents of A (B).

Registers Affected: A, B Timing: 13

Example: Assume
(A) = 0027
(B) = 4335
(1000) = 0036

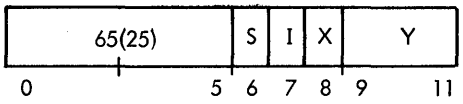
Performing DVA 01000 yields
(B) = 6217
(A) = 0033

The division is performed as follows:



LOGICAL INSTRUCTIONS

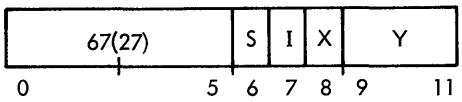
ANA (ANB) AND TO A (AND TO B)



ANA performs a logical AND with the contents of the A Register and the contents of the effective memory location; it places the result in A. The previous contents of A are lost.

Registers Affected: A, (B) Timing: 2

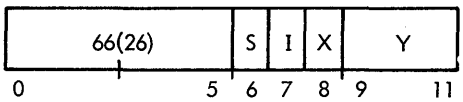
ORA (ORB) OR TO A (OR TO B)



ORA performs a logical "inclusive OR" with the contents of the A Register and the contents of the effective memory location; it places the result in A. The previous contents of A are lost.

Registers Affected: A, (B) Timing: 2

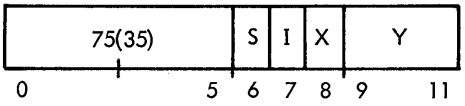
EOA (EOB) EXCLUSIVE OR TO A (EXCLUSIVE OR TO B)



EOA performs a logical "exclusive OR" with the contents of the A Register and the contents of the effective memory location; it places the result in A. The previous contents of A are lost.

Registers Affected: A, (B) Timing: 2

MAA(MAB) MEMORY AND A TO MEMORY (MEMORY AND B TO MEMORY)

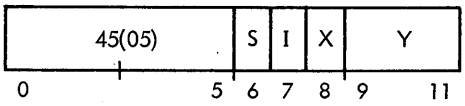


MAA performs a logical AND with the contents of the A Register and the contents of the effective memory location; it places the result in the effective memory location. The previous contents of the memory location are lost; MAA leaves the contents of A undisturbed.

Registers Affected: M Timing: 3

COMPARISON INSTRUCTIONS

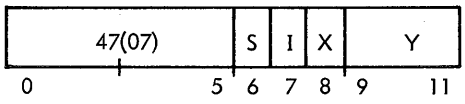
COA(COB) COMPARE ONES WITH A (COMPARE ONES WITH B)



COA compares the contents of the A Register, bit by bit, with the contents of the effective memory location. If the contents of A and the contents of the effective location have any ones in corresponding bit positions, COA resets the Flag Bit. If there is no such corresponding pair of one bits, COA sets the Flag Bit.

Registers Affected: F Timing: 2

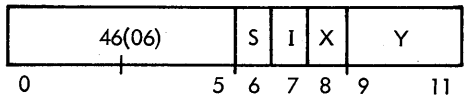
CMA(CMB) COMPARE MAGNITUDE OF M WITH A (COMPARE MAGNITUDE OF M WITH B)



If the contents of the A Register are arithmetically less than the contents of the effective memory location, CMA resets the Flag Bit. Otherwise, it sets the Flag Bit.

Registers Affected: F Timing: 2

CEA(CEB) COMPARE M EQUAL TO A (COMPARE M EQUAL TO B)

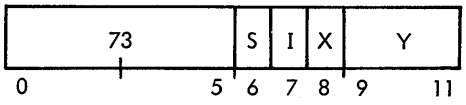


If the contents of the A Register are equal to the contents of the effective memory location, CEA resets the Flag Bit. Otherwise, it sets the Flag Bit.

Registers Affected: F Timing: 2

BRANCH INSTRUCTIONS

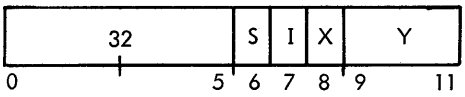
BRU BRANCH UNCONDITIONALLY



The computer takes the next instruction from the location determined by the effective address. BRU cannot be interrupted.

Registers Affected: None Timing: 1

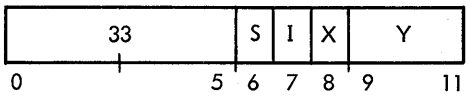
BRC BRANCH, CLEAR INTERRUPT AND LOAD FLAG



The computer takes the next instruction from the location determined by the effective address; it also clears the currently active interrupt level. If BRC uses direct addressing, it clears the Flag Bit and sets the PCT bit. If it uses indirect addressing, BRC places into the Flag Bit and PCT bit the content of bits 0 and 1 of the first word of the last indirect address pair.

Registers Affected: F, PCT Timing: 3

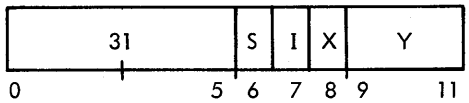
BRL BRANCH AND LOAD FLAG



BRL transfers to the effective memory location. If BRL uses direct addressing, it clears the Flag Bit and sets the PCT bit. If it uses indirect addressing, BRL places into the Flag Bit and the PCT bit the content of bits 0 and 1 of the first word of the last indirect address pair. BRL cannot be interrupted.

Registers Affected: F, PCT Timing: 1

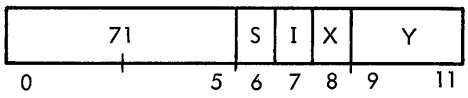
BFF BRANCH ON FLAG FALSE



If the content of the Flag Bit is zero, the computer takes the next instruction from the location determined by the effective address. If the content is one, the computer executes the next instruction in sequence. If a branch occurs, there can be no interrupt.

Registers Affected: None Timing: 1 if Branch
2 if No Branch

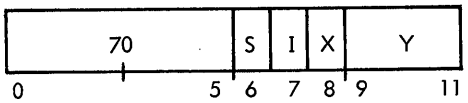
BFT BRANCH ON FLAG TRUE



If the content of the Flag Bit is one, the computer takes the next instruction from the location determined by the effective address. If the content is zero, the computer executes the next instruction in sequence. If a branch occurs, there can be no interrupt.

Registers Affected: None Timing: 1 if Branch
2 if No Branch

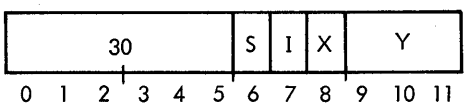
BDA BRANCH ON DECREMENTING A



BDA decrements the contents of the A Register by one. It then tests the result unequal to 7777_8. If unequal, the computer takes the next instruction from the location determined by the effective address. If equal, the computer executes the next instruction in sequence.† If a branch occurs, there can be no interrupt. If this instruction specifies indexing, the indexing is performed before A (the index register) is decremented.

Registers Affected: A Timing: 1 if Branch
2 if No Branch

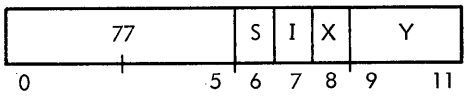
BAX BRANCH AND EXCHANGE A AND B



BAX exchanges the contents of A with the contents of B; then it branches to the location determined by the effective memory address. BAX cannot be interrupted. If this instruction specifies indexing, the indexing will be performed before the interchange of "A" and "B."

Registers Affected: A, B Timing: 1

BRM BRANCH AND MARK PLACE

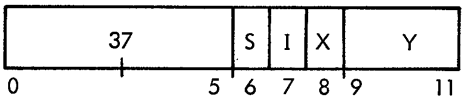


BRM stores the contents of the Program Counter (which contains the address of the next instruction in sequence) in bits 9 through 11 of the effective memory location and bits 0 through 11 of the effective location plus one. It stores the content of the Flag Bit in bit 0 and the content of the PCT bit in bit 1 of the effective location; bits 2 through 5 of the effective location are unpredictable. Bits 6 through 8 of the effective location are cleared. BRM then branches to the effective memory location plus two. Immediate addressing is not allowed.

Registers Affected: M, M+ 1 Timing: 3

†As with any instruction, when using immediate addressing with a BDA, the phrase "the next instruction in sequence" refers to the instruction two locations beyond the BDA.

BMC BRANCH, MARK PLACE, AND CLEAR FLAG



BMC stores the contents of the Program Counter in bits 9 through 11 of the effective memory location and bits 0 through 11 of the effective location plus one. It stores the contents of the Flag Bit in bit 0 and the contents of the PCT bit in bit 1 of the effective location; bits 2 through 5 of the effective location are unpredictable. Bits 6 through 8 of the effective location are cleared. The Flag Bit is cleared and the PCT bit is set. BMC then branches to the effective memory location plus two. Immediate addressing is not allowed.

Registers Affected: M, M + 1, F, PCT Timing: 3

Note that the BMC instruction is the one normally executed when an interrupt occurs. The address stored in this case is the location of the next instruction to be executed in the main program.

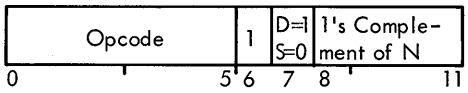
SHIFT INSTRUCTIONS

Shift instructions operate on the A, B, and Flag Registers. The shifts can be single or double register. All shifts are to the left. The number of shifts N is specified in the least significant 4 bits of the effective address. The maximum number of shifts is 15 (17_8); zero is allowed. N is written in the 4 bits, in one's complement form (i.e., a shift N=7 appears as 10_8).

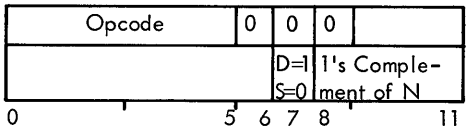
The single or double shift is determined via bit 7 of the effective address; it is a 0 for single-register shift and a 1 for double-register shift.

The conventional address formats are:

One-Word Address



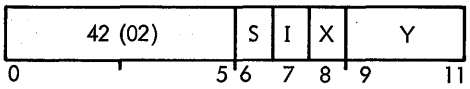
Two-Word Address



Shift Timing

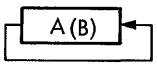
Shift Count (Decimal)	Timing (Cycles)
0 - 3	3
4 - 6	4
7 - 9	5
10 - 12	6
13 - 15	7

CYA(CYB) CYCLE A (CYCLE B)

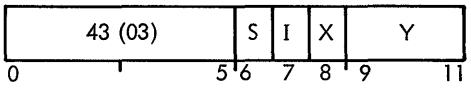


CYA shifts the contents of the A Register N places to the left. All bits shifting past position 0 shift into position 11. The one's complement of N, the number of positions to be shifted, is placed in the least significant 4 bits of the effective address.

Registers Affected: A(B)

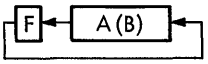


CFA(CFB) CYCLE FLAG AND A (CYCLE FLAG AND B)

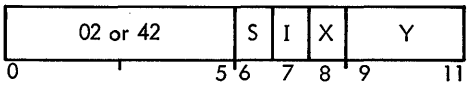


CFA shifts the contents of the A Register and the Flag Bit, taken as a single 13-bit register, N places to the left. All bits shifting past position 0 shift into the Flag Bit, bits from the Flag Bit go into position 11. The one's complement of N is placed in the least significant 4 bits of the effective address.

Registers Affected: A(B), F

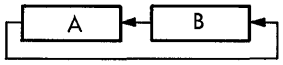


CYD CYCLE DOUBLE

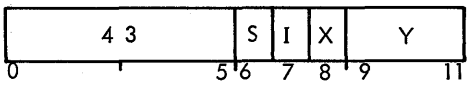


CYD shifts the contents of the A and B Registers N places to the left. All bits shifting out of position 0 of A shift into position 11 of B; all bits shifting out of position 0 of B shift into position 11 of A. The one's complement of N is placed in the least significant 4 bits of the effective address.

Registers Affected: A, B

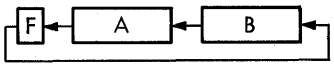


CFD CYCLE FLAG AND DOUBLE

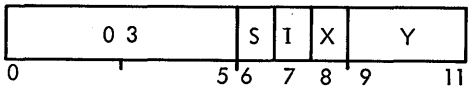


CFD shifts the contents of the A and B Registers and the Flag Bit, taken as a single 25-bit register, N places to the left. Bits shift from position 0 of B into position 11 of A, from position 0 of A into the Flag Bit, and from the Flag Bit into position 11 of B. The one's complement of N is placed in the least significant 4 bits of the effective address.

Registers Affected: A, B, F

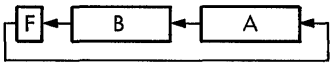


CFI CYCLE FLAG AND DOUBLE INVERSE



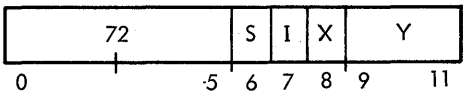
CFI shifts the contents of the B and A Registers and the Flag Bit, taken as a single 25-bit register, N places to the left. Bits shift from position 0 of A into position 11 of B, from position 0 of B into the Flag Bit, and from the Flag Bit into position 11 of A. The one's complement of N is placed in the least significant 4 bits of the effective address.

Registers Affected: A, B, F



CONTROL INSTRUCTIONS

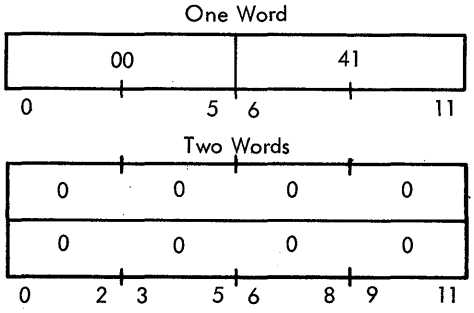
EXU EXECUTE



EXU executes the instruction in the effective memory location. Then the computer executes the next instruction following EXU. If the effective memory location contains a branch, control goes to the branch-to location. If the effective location contains another EXU, the process repeats with control always returning to the next location after the first EXU or to the branch-to location if the last instruction is a branch instruction. Immediate addressing is not allowed on EXU (this restriction does not apply to the instruction executed). No "trappable" instruction that will trap can be executed.

Registers Affected: None Timing: 1 + time of instruction execution

HLT HALT



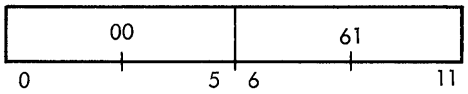
HLT halts instruction execution and lights the HALT light. To resume computation, the operator sets the mode switch to IDLE and then to RUN or STEP at which time the computer executes the instruction in the location addressed by the contents of the Program Counter. (This will be the instruction following the HLT instruction if the operator has not changed the contents of the Program Counter.)

If an interrupt occurs while halted by a HALT (while still in RUN), the computer acknowledges the interrupt and computation resumes. (The instruction following the HLT instruction will be executed following the processing of the interrupt.)

Registers Affected: None Timing: 3, 4

TRAPPING INSTRUCTIONS

SCT SET PROGRAM-CONTROLLED TRAP

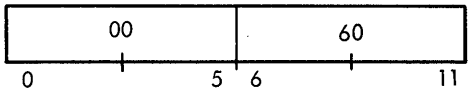


SCT unconditionally sets the PCT bit to a one.

SCT cannot be interrupted.

Registers Affected: PCT Timing: 3,4

RCT RESET PROGRAM-CONTROLLED TRAP

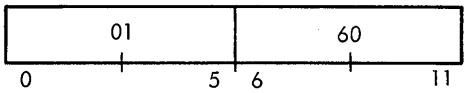


RCT unconditionally sets the PCT bit to a zero.

RCT cannot be interrupted.

Registers Affected: PCT Timing: 3,4

TCT TEST PROGRAM-CONTROLLED TRAP



TCT tests the status of the PCT bit. If PCT is a one, it sets the Flag Bit to a one. If PCT is a zero, it sets the Flag Bit to a zero.

Registers Affected: F Timing: 3,4

BREAKPOINT TESTS

This instruction tests the status of the Breakpoint switches. If the selected switch is set, the Flag Bit is set to a 1. If the switch is reset, the Flag is set to a 0.

Mnemonic	Coding
BPT 1	0144
BPT 2	0145
BPT 3	0146
BPT 4	0147

Registers Affected: F Timing: 3,4

III. INPUT/OUTPUT INSTRUCTIONS

INTRODUCTION

The SDS 92 has a comprehensive input/output system to complement its high internal processing speed. This system can transmit data in word, character, and single-bit form to and from the computer at the speed of internal computation. The input/output system is of great variety and can assume control of conditions imposed by a wide range of input/output, special system devices; but the computer always leaves a high degree of flexible input/output control to the programmer.

The system contains:

Buffered input/output of data words, under program control in blocks or as single words.

Input/output of blocks of data via subchannels; up to 64 channels simultaneously operating through a multiplexing system (optional).

Direct parallel input/output of 12-bit words, singly or in blocks, to and from external static registers.

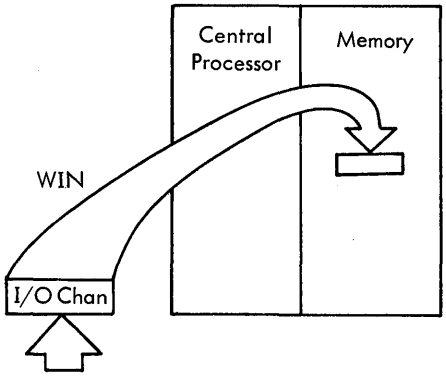
Single bit input/output, such as equipment on/off status, sense switches, and pulsing and sending of special signals.

I/O CHANNEl

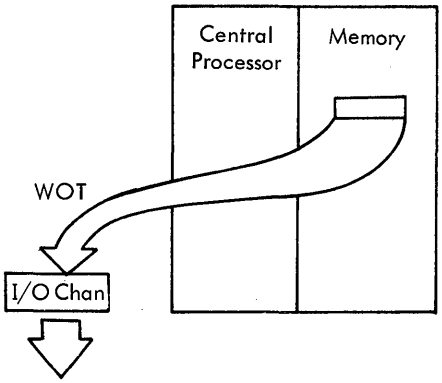
The I/O Channel, standard equipment in the computer, performs input/output of words singly or in blocks. On output, the I/O channel transmits words in 6-bit characters, one or two characters per word as selected, or in 12-bit (optional) single character form. On input, the I/O channel receives words in 6-bit characters, one or two per word, or 12-bit (optional) characters as desired. This channel transmits single words or blocks. The program places the block count in the A Register prior to the transfer and the computer counts this down to 7777 automatically to terminate the transfer operation.

SINGLE WORD TRANSMISSION

Using the I/O Channel, a program can transmit data words between memory and peripheral devices under single instruction control. To do this, the program first activates the channel and the peripheral device with an energize or "alert" instruction (one of the configurations of the multi-purpose instruction, ENERGIZE OUTPUT M (EOM)). WOT is the WORD OUT instruction; WIN is the WORD IN instruction. WIN causes a word from a peripheral device to be taken from the I/O channel and placed directly into the specified memory location without disturbing any internal registers.



WOT causes a word to be taken from the specified memory location and placed in the I/O Channel to be output when requested by the currently active peripheral device.



To transfer blocks of data words via the I/O channel, the program uses the same EOM configuration to set the channel for operation; the program specifies the number of words in the transfer by placing the word count minus one in the A Register ((A) + 1 = N). The RECORD OUT (ROT) instruction causes the computer to output words from the effective address M through the effective address plus the contents of A (M through M + (A)). The RECORD IN (RIN) instruction causes the computer to input from the actively transmitting peripheral device; the operation terminates when the computer receives N words, or when it receives an "end-of-record" from the peripheral device. RIN and ROT tie up the computer during the entire input/output transmission.

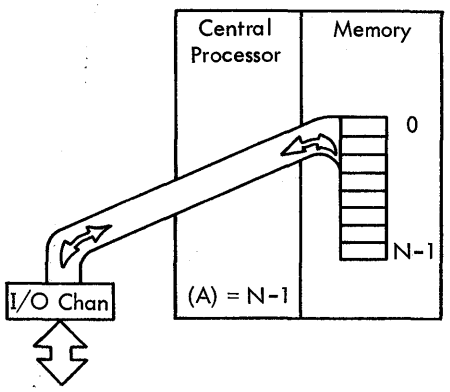


Figure 3-1. ROT/RIN Data Transmission

Two instructions control the process of transmitting and receiving data between peripheral equipment and the central processor using the I/O Channel. These instructions are:

EOM	ENERGIZE OUTPUT M
SES	SENSE EXTERNAL SIGNAL

The EOM instruction activates the I/O Channel, selects the peripheral device, and requests the desired operation. The programmer uses the SES instruction to test for all input/output operational conditions; SES is multipurpose like the EOM. Later sections describe the exact configurations of EOM and SES.

DIRECT PARALLEL INPUT/OUTPUT, 12 BIT

The parallel input/output facility allows full, 12-bit words to be transmitted directly out of and into the memory. After activating the peripheral device or special system device with an activating EOM, the PARALLEL OUTPUT (POT) and PARALLEL INPUT (PIN) cause any selected word in core memory to be presented to the peripheral device connector; or conversely, cause a word (12-bit signal) received into the device connector to be stored in the selected location. POT/PIN also check or generate correct memory parity with each word transmitted. The system provides a block transfer form of POT and PIN with the instructions, BLOCK PARALLEL IN (BPI) and BLOCK PARALLEL OUT (BPO). By placing the word count N minus one in the A Register, BPI and BPO provide the identical function of PIN and POT, respectively, on N consecutive words.

Parity checking/generating is automatic for these operations on machines equipped with the memory parity feature.

See POT/BPO, PIN/BPI Instructions in this section.

DIRECT PARALLEL INPUT/OUTPUT, 24-BIT (Optional)

A 12-bit register is available to extend the word for POT/PIN operations to 24 bits. For a POT, the device operates as follows: EOM to activate the extender, POT to place the most significant 12 bits in the extender, EOM to activate the external device to get the data, and POT to transmit the lower 12 bits. This last POT transmits the entire 24 bits.

For a PIN, EOM alerts the sending device, PIN stores the least significant 12 bits, the high-order bits fill the extender, EOM alerts the extender, and another PIN stores the contents of the extender. Neither of these coding sequences can be interrupted. See the Interrupt System paragraphs for the Alert Extender EOM for both POT and PIN.

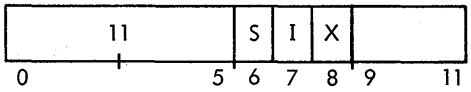
SINGLE BIT INPUT/OUTPUT

The EOM and SES instruction provide a general single bit transmitting and sensing facility for use as test and control signals with special systems and standard peripheral devices.

DATA TRANSFER INSTRUCTIONS

No interrupt can occur between any of these instructions and the instruction following it.

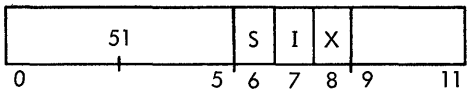
WOT WORD OUT



WOT transfers the contents of the effective memory location into the I/O channel buffer. If the buffer is not ready, the central processor "hangs up" until the buffer empties from a previous instruction and is ready to accept the new data word.

Registers Affected: None Timing: 4 + wait

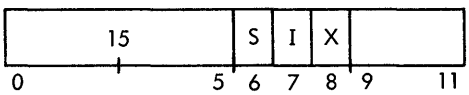
ROT RECORD OUT



Starting with the effective memory location, ROT transfers N sequential words into the I/O channel buffer. The contents of the A Register are the word count N minus one; ROT can output to 4096 words per execution. The central processor must wait as with WOT before it transfers the first word to the buffer; it also must wait for the buffer to clear between each word transfer. ROT completely ties up the computer until the Nth word transfers into the channel buffer. The next instruction executes before the Nth word transfers out of the channel buffer to the connected peripheral device.

Registers Affected: A Timing: 2 + 2N + wait

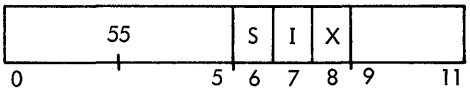
WIN WORD IN



WIN transfers the contents of the channel buffer into the effective memory location. If the buffer is not already filled, the central processor "hangs up" until the buffer fills with the word being received from the peripheral device.

Registers Affected: M Timing: 5 + wait

RIN RECORD IN



Starting with the effective memory location, RIN transfers N words from the channel buffer into sequential locations. The contents of the A Register are the word count N minus one; RIN can input up to 4096 words per execution. The central processor must wait as with WIN before it receives the first word from the channel buffer; it also must wait for the buffer to fill between each word transfer. RIN completely ties up the computer until the Nth word is in memory. This input will also terminate if the computer receives an END signal from the peripheral device before the Nth word. In either case, the computer executes the next instruction after RIN terminates.

Registers Affected: M to M + (A), A Timing: 3 + 2N + wait

I/O CHANNEL OPERATION

The I/O Channel can control up to 30 input/output devices; it automatically handles character, word assembly/disassembly, and input/output parity detection and generation.

The channel is bi-directional and communicates with 6-bit character devices (12-bit optional). The program specifies whether one or two characters are to be assembled/disassembled in each word during the transmission.

The program uses a Buffer Control EOM to set the operation controls such as forward/backward tape direction, to place the unit address in the channel, and to initiate the proper assembly/disassembly mode. The presence of the unit address activates the channel causing it to look for data coming from a peripheral device or from memory, as determined by the unit address (see the Unit Address Code, Table 3-1).

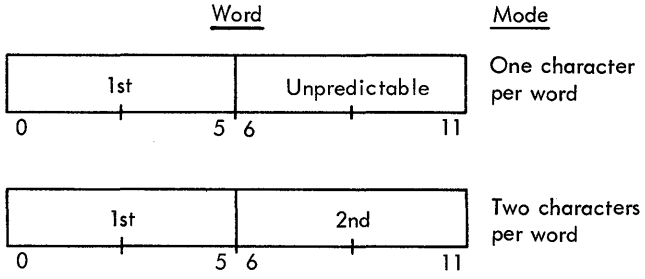
To get data from the channel buffer after it is received there from a peripheral device during input, the program uses a WORD IN (WIN) instruction, or its block transfer equivalent, RIN. To place data into the channel buffer so that the channel can transmit it to the waiting peripheral device, the program uses WORD OUT (WOT), or its block transfer equivalent, ROT.

I/O CHANNEL BUFFER DESCRIPTION (See Figure 3-2.)

During the execution of ROT/RIN, the computer is completely tied up while it handles the data transfers, increments the memory location address for the data transfers, and tests for transfer termination using the word count N (by decrementing A by one whenever a word is transferred).

Each of the 30 devices which can be attached to a buffer has a unique, two-digit, octal address by which it is chosen for an input/output operation. To choose the peripheral device, the program loads the proper unit address into the 6-bit Unit Address Register (UAR). This address selects both the device and, if appropriate, the function to be performed. Placing a non-zero unit address in the Unit Address Register "connects" the peripheral unit addressed to the buffer and the buffer becomes "active". When the UAR contains a zero address, or any time that a terminal or initial condition clears the contents of UAR, the buffer is "inactive", and it is not connected to a peripheral unit.

The Word Assembly Register (WAR) and the Single Character Register (SCR) comprise the active portion of a buffer. The Word Assembly Register, a 12-bit, word-sized buffer, contains the word of data actively being received or transmitted during an input or output operation. During input, 6-bit characters (plus parity) come into the Single Character Register where the channel assembles them, one at a time, into the WAR. Depending on the number of characters per word specified, the word assembled during input has the form:



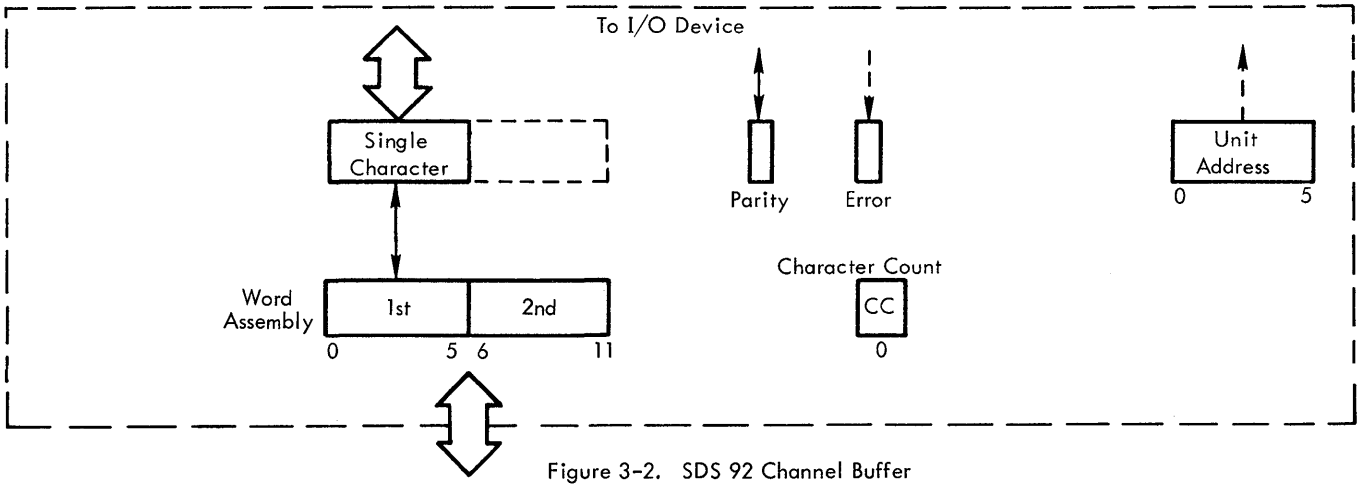


Figure 3-2. SDS 92 Channel Buffer

Table 3-1. Unit Address Codes

00	Disconnect	40	-
01	Type Input No. 1	41	Type Output No. 1
02	Type Input No. 2	42	Type Output No. 2
03	Type Input No. 3	43	Type Output No. 3
04	Paper Tape Input No. 1	44	Paper Tape Punch Output No. 1
05	Paper Tape Input No. 2	45	Paper Tape Punch Output No. 2
06	Card Reader Input No. 1	46	Card Punch Output No. 1
07	Card Reader Input No. 2	47	Card Punch Output No. 2
10	Magnetic Tape Input No. 0	50	Magnetic Tape Output No. 0
11	Magnetic Tape Input No. 1	51	Magnetic Tape Output No. 1
12	Magnetic Tape Input No. 2	52	Magnetic Tape Output No. 2
13	Magnetic Tape Input No. 3	53	Magnetic Tape Output No. 3
14	Magnetic Tape Input No. 4	54	Magnetic Tape Output No. 4
15	Magnetic Tape Input No. 5	55	Magnetic Tape Output No. 5
16	Magnetic Tape Input No. 6	56	Magnetic Tape Output No. 6
17	Magnetic Tape Input No. 7	57	Magnetic Tape Output No. 7
20	-	60	High-Speed Printer Output No. 1
21	-	61	High-Speed Printer Output No. 2
22	-	62	-
23	-	63	-
24	-	64	Incremental Plotter Output No. 1
25	-	65	Incremental Plotter Output No. 2
26	Disc File Input No. 1	66	Disc File Output No. 1
27	Disc File Input No. 2	67	Disc File Output No. 2
30	Scan Magnetic Tape No. 0	70	Magnetic Tape Erase No. 0
31	Scan Magnetic Tape No. 1	71	Magnetic Tape Erase No. 1
32	Scan Magnetic Tape No. 2	72	Magnetic Tape Erase No. 2
33	Scan Magnetic Tape No. 3	73	Magnetic Tape Erase No. 3
34	Scan Magnetic Tape No. 4	74	Magnetic Tape Erase No. 4
35	Scan Magnetic Tape No. 5	75	Magnetic Tape Erase No. 5
36	Scan Magnetic Tape No. 6	76	Magnetic Tape Erase No. 6
37	Scan Magnetic Tape No. 7	77	Magnetic Tape Erase No. 7

An unfilled character position is unpredictable. When assembled during a single-word operation, a WIN instruction places the word into memory. With RIN, the computer places each word in memory when assembled.

During output, words come from memory into the WAR where the channel disassembles them into the SCR, one 6-bit character at a time. Depending on the characters per word mode specified, the channel transmits the 6-bit characters (with generated parity) as follows:

Function	Mode
Output one character from bits 0 through 5	One character per word
Output two characters from bits 0 through 5, 6 through 11	Two characters per word

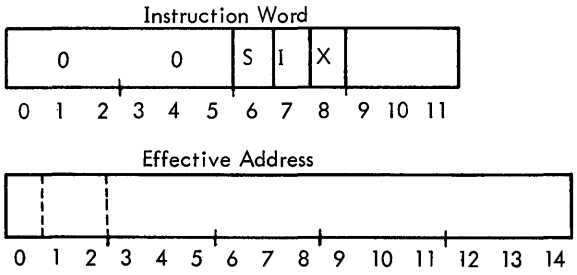
After the first character transfer, the word in the WAR shifts left six bits to be ready for the next transfer, when two characters from each word are used. Under ROT control, a new word containing the next character(s) comes to the WAR when it is required.

EOM INSTRUCTIONS (opcode 00)

BASIC CONFIGURATION

The EOM instruction is a multipurpose instruction that operates in four distinct modes with many functional configurations. The modes are Buffer Control, Input/Output Control, Internal Function Control, and System Control.

EOM ENERGIZE OUTPUT M



The EOM uses the 15 bit configurations of the effective memory address as a control word to select the different control modes and to select all additional control functions. EOM allows all addressing modes in obtaining the effective address.

Setting the two bits (1, 2) in the address determines the mode of the EOM:

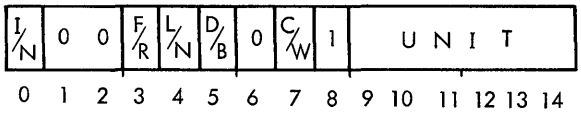
1	2	Control Mode
0	0	Buffer
0	1	Input/Output
1	0	Internal
1	1	System

The Buffer Control EOM operates essentially as a setup or preparation facility for data transmissions or other peripheral functions using the I/O Channel. The Input/Output Control EOM directs peripheral devices directly in such operations as rewind tape and upspace the printer. EOM in the Internal Control mode performs internal control operations such as activating the (optional) 24-bit PIN/POT extender logic. The System EOM is specifically concerned with special systems; the system determines the particular uses. EOM in any of the last three modes also can alert a device for a POT or PIN type operation.

NOTE: If an interrupt occurs during the execution of an EOM, no acknowledgement occurs until the completion of the execution of the instruction following the EOM.

Registers Affected: None Timing: 3,4

BUFFER CONTROL EOM (effective address)



Designation	Function
I/N	Bit position 0 specifies Interlace operation. A "0" specifies no Interlace operation. A "1" alerts the Interlace.
00	Bit positions 1 and 2 contain the EOM mode indicator for the Buffer Control mode.
F/R	Bit position 3 specifies the direction in which the peripheral device operates. A "0" specifies the forward direction. A "1" specifies the reverse direction.
L/N	Bit position 4 specifies whether the device should be started with a leader as in paper tape. A "0" specifies a start with leader. A "1" specifies a start without leader.
D/B	Bit position 5 specifies the mode of character format. A "0" specifies BCD format. A "1" specifies Binary format. When this is not appropriate, bit 5 provides special control.
0	Bit position 6 is unassigned.
C/W	Bit position 7 specifies the number of characters to be assembled into, or disassembled from, each transmitted word. 0 specifies one character per word, 1 specifies two. One character per word, 0, is used for full-word (12-bit characters) transmission (optional).
1	Bit position 8 must always be 1.
UNIT	Bit positions 9 through 14 specify the unit and the function to be performed with that unit.

INPUT/OUTPUT CONTROL EOM (effective address)

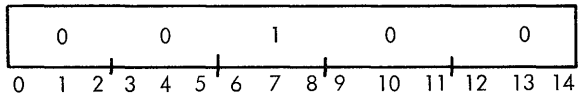
Designation	Function
I/N	Bit 0 specifies Interlace operation. A "0" specifies no Interlace. A "1" alerts the Interlace.
01	Bits 1, 2 specify the Input/Output Control mode.
FUNCTION	Bits 3 through 7 specify control peculiar to each device.
1	Bit 8 must be 1.
UNIT	Bits 9 through 14 contain the Unit Address of the specified device.

A Unit Address of 00 refers to the I/O Channel itself.

STANDARD EOM INSTRUCTIONS

These EOM effective address configurations have standard uses.

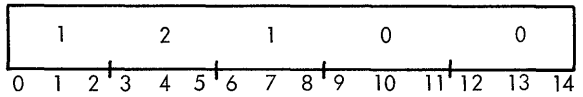
DSC DISCONNECT CHANNEL



DSC disconnects the I/O Channel. This instruction unconditionally sets the UNIT Address Register to 00 regardless of whether the channel is currently addressing a device. DSC disconnects any device connected to the channel; it unconditionally makes the channel inactive and clears the error indicator.

Registers Affected: None Timing: 3, 4

TOP TERMINATE OUTPUT ON THE CHANNEL

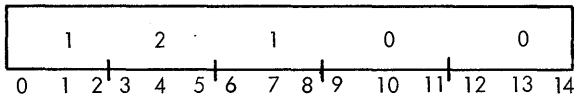


During output when the last word of a block goes to the channel, TOP terminates output. After execution of TOP, the following occurs. When the channel delivers the last character to the peripheral device, the channel disconnects.

TOP must always terminate a channel output operation.

Registers Affected: None Timing: 3, 4

TIP TERMINATE INPUT ON THE CHANNEL



During input when the last (desired) word has been stored in memory, TIP terminates input.

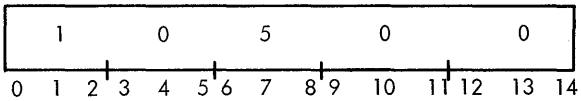
TIP or DSC should always terminate a channel input operation.

Registers Affected: None Timing: 3, 4

After TIP is given during an input operation, the following occurs:

1. The I/O channel receives any further characters from the input device — as before.
2. All error checks are performed — as before.
3. However, the Word Assembly Register is never again considered "full". This means:
   a. Interlace operations store no more words.
   b. Non-interlace operations give no more End-of-Word (I1) Interrupts.
4. The above "scanning-type" sequence continues until the End-of-Record at which time:
   a. The End-of-Record (I2) Interrupt is sent (if armed).
   b. The channel disconnects.
   c. The channel becomes inactive.

ASC ALERT TO STORE INTERLACE COUNT



ASC alerts the interlace option that the PIN to follow is a request for the contents of the current COUNT contents. The sequence:

```
ASC
PIN
```

stores the current contents of the COUNT register into location M. See Interlace Option, this section.

NOTE: The above sequence must be consecutive; no other instruction should be interposed.

Registers Affected: None Timing: 3, 4

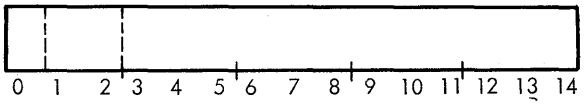
SES INSTRUCTIONS (opcode 01)

BASIC CONFIGURATION

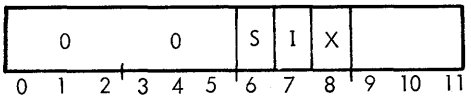
The SES is a multipurpose test instruction used for testing responses to the input/output channel and attached peripheral devices as well as for testing internal and external indicators.

SES SENSE EXTERNAL SIGNAL

Instruction Word



Effective Address



Like the EOM, the SES uses the bit configuration of the effective address to select the different tests and also operates in four modes that are selected by address bits 1 and 2:

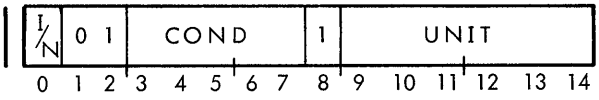
1	2	Test Mode
0	0	Buffer
0	1	Input/Output
1	0	Internal
1	1	System

When executed, an SES tests for a specified condition and sets or resets the Flag Bit in response to the condition. The program determines the Flag Bit status via one of the branch-on-flag instructions. SES allows all addressing modes in obtaining the effective address.

The Buffer and Input/Output Test SESs are the complement of the Buffer and Input/Output Control EOMs; they sense

the conditions of the I/O Channel and its connected peripheral devices.

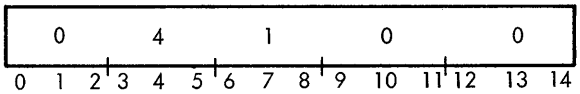
INPUT/OUTPUT TEST SES (effective address)



Designation	Function
I/N	Bit 0 specifies Interlace operation. A "0" specifies no Interlace. A "1" alerts the Interlace.
01	Bits 1 and 2 specify the Input/Output Test mode.
COND	Bits 3 through 7 specify conditions to be sensed.
1	Bit 8 must be 1.
UNIT	Bits 9 through 14 contain the Unit Address of the specified device.

STANDARD BUFFER SES INSTRUCTIONS (effective address)

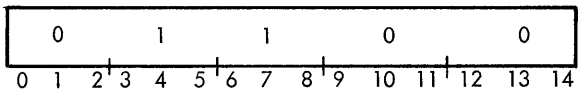
CAT CHANNEL ACTIVE TEST; SET FLAG IF NOT ACTIVE



If the channel is ready to accept a new input/output instruction, CAT sets the Flag Bit. If the channel is active, CAT resets the Flag Bit. (The channel will test active during an input operation - even after the peripheral has terminated its operation - until all meaningful data words in the character buffer have been stored in memory.)

Registers Affected: F Timing: 3, 4

CET CHANNEL ERROR TEST; SET FLAG IF ERROR



CET tests the error indicator in the I/O Channel for being set. If set to no error, CET resets the Flag Bit. If set to error, CET sets the Flag Bit.

Registers Affected: F Timing: 3, 4

INTERLACE

The I/O Channel interlace is an optional hardware device that can control and perform input/output operations independent of, and simultaneous with, central processor program execution. In using the interlace, the program sets up a starting address for data in or out, sets up a record length of the data to be read or written, and starts the interlaced operation. The program then continues computation while the interlace monitors the I/O operation, accessing memory when necessary, incrementing the data address as needed, and counting the number of words in the record. Usually, when the count goes through zero, the operation is completed and the program can use the newly entered data and/or can reset the Interlace for another independent, I/O transmission.

The Interlace contains two registers: the 12-bit COUNT register to contain the record count and the 15-bit ADDRESS register to contain the data address. When loading the record count (N), the program places N-1 (one less than the record count) in the COUNT register.

The program can use any I/O or Buffer Control EOM to "alert" the Interlace for operation. A 1-bit in bit positions 0 and 8 of the effective address generated by the EOM will alert the Interlace.

The standard assembler form for alerting the Interlace is an asterisk (*) in the first column of the address of the EOM that activates the peripheral device in the I/O transmission. For example, to alert the Interlace while activating the magnetic tape unit number 1 to read tape decimal, one could write:

```
RTD *1, 2
```

A special EOM to alert only the Interlace,

```
ALC     ALERT CHANNEL INTERLACE
```

has the address form 50100; no "*" is needed.

The PARALLEL OUTPUT (POT) instruction transmits the starting address and record count to the Interlace. The three "POTs" that are needed function as follows (RTD *1, 2 is used as the alerting EOM):

```
RTD     *1,2
POT     HIADDR
POT     LOWADDR
POT     NLESSONE
```

(NOTE: This sequence must be consecutive; no other instructions should be interposed.)

where:

The least significant bits of the contents of location HIADDR form the high or most significant three bits of the starting address,

the contents of location LOWADDR form the least significant 12 bits of the 15-bit starting address, and

the contents of NLESSONE are the record length minus one (N-1).

In each case, the POT transmits the information into the proper Interlace register.

The termination of an interlace operation can be determined by using a CAT, CHANNEL ACTIVE TEST; or the progress of an operation may be monitored through the use of the ALERT TO STORE INTERLACE COUNT.

ASC (10500) alerts the Interlace that a PIN is to follow to get a record of the current COUNT contents.

The sequence:

```
ASC
PIN M
```

alerts the Interlace and "PINs" the current contents of the COUNT register into location M. The contents of COUNT are N-1 minus the number of words of the record already transmitted.

I/O CHANNEL INTERRUPTS OPTION

Two interrupts I1 and I2 are directly associated with the I/O channel. These are priority interrupts with I1 having priority over I2.

When I1 is requested, it interrupts each time the I/O channel buffer empties or fills; that is, when requested it occurs on input each time the buffer collects a word, or it occurs on output each time the buffer transmits a complete word.

When I2 is requested, it interrupts when an End-of-Record occurs; that is, it interrupts only after a complete record is input or output.

I1 and I2 are always enabled (as described in the Interrupt Paragraphs, Input/Output Section of this manual).

A special instruction, EOM 11X00, arms/disarms these interrupts. The values of X are:

X	Function
1	Disarm I1 and I2
3	Arm I1, disarm I2
5	Arm I2, disarm I1
7	Arm I1 and I2

The instruction EOM 13X00, as a terminate output EOM, can be used effectively in conjunction with the arming feature. For instance, EOM 13500 terminates output, arms I2 and disarms I1. It functions like this: When the current output from the I/O channel is finished and the I/O buffer is free, the I2 interrupt occurs.

The standard assembler mnemonic and instruction form is:

```
ARM   X
```

where X is as described above.

Interrupts Used with the Interlace Option

During Interlace operation, the basic interrupts function according to the names they are given below:

I1 is COUNT EQUAL 7777_8
I2 is END OF RECORD

When requested, I1 occurs when COUNT goes through zero. When requested, I2 occurs when an End-of-Record occurs.

On output or input:

If I1 is not armed, the Interlace terminates the channel (i.e., effects an automatic TOP or TIP) when the COUNT goes through zero.

Note: During input, this means that the peripheral continues to the end-of-record, but no more input words are stored in memory.

If I1 is armed, the Interlace does not terminate the channel on COUNT passing zero; instead an I1 interrupt is generated. This allows the program to re-initiate the COUNT and starting ADDRESS in the Interlace and continue performing the same I/O operation.

The Channel always disconnects when the end-of-record occurs with no regard to the interrupt arms.

Note: When armed by ARM X, an interrupt condition occuring on I1 or I2 causes the interrupt level to go to the Waiting state. If the Interrupt System is Enabled, the respective interrupt will go to the Active state as its priority permits. If the Interrupt System is Disabled, the interrupt stays in the Waiting state indefinitely.

POT/BPO, PIN/BPI INSTRUCTIONS

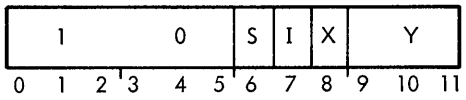
Two instructions, PARALLEL OUTPUT (POT) and PARALLEL INPUT (PIN), cause any word in core memory to be presented in parallel at a connector; or, inversely, cause signals sent to a connector to be stored in any core memory location. The execution of a POT or PIN instruction causes a signal to be sent to the external device involved in the input/output operation. During a PIN, this signal tells the device to send its data word as soon as it is operational. Wehn a device becomes operational during a READ or PIN operation, it transmits a Ready signal to the central processor while at the same time presenting its data word. The computer places the received data word into a specified memory location without disturbing any arithmetic registers. The computer "hangs up" during the execution of PIN until it receives the Ready signal from the external device.

During the execution of a POT instruction, the central processor transmits a signal to the external device alerting it to receive a data word. When the device becomes operational, it transmits a Ready signal to the central processor which releases the data word to the external device. The computer "hangs up" during the execution of POT until it receives the Ready signal from the external device. The block transfers forms of these instructions are BLOCK PARALLEL INPUT (BPI) and BLOCK PARALLEL OUTPUT (BPO).

Special system requirements demand that complete words of control or data information be transferred between the central processor and the special external devices. The PIN or POT preceded by the activating EOM gives exactly this facility. The EOM alerts the system device by specific address and the PIN or POT transfers the required word. That is, the EOM activates and alerts the special device and the PIN/POT transfers 12 bits to or from the effective memory location specified. To avoid a posssible computer "hang-up", the SES instruction can test the Ready signal of the special device prior to the EOM and PIN/POT. If the Ready signal from the external device sets one of the priority interrupts (optional), parallel input/output operation can occur as soon as the external device is able to transmit or receive. Since the Ready signal initiating the interrupt persists through the POT or PIN execution, no "hang-up" occurs.

No interrupt can occur during the execution of, or between any of these instructions and the instruction following it.

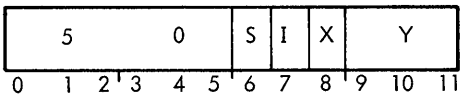
POT POT PARALLEL OUTPUT



POT transmits the contents of the effective memory location in parallel to 12 output lines of an external device.

Registers Affected: None Timing: 4 + wait (high-speed) and 4, 5 + wait

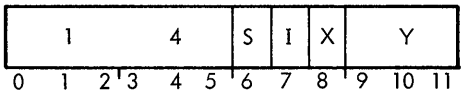
BPO BLOCK PARALLEL OUTPUT



Starting with the effective memory location, BPO transfers the contents of N sequential locations in parallel to 12 output lines of an external device. The contents of the A Register are the word count N minus one; BPO can output up to 4096 words per execution. The output lines fill and empty under control of BPO and the Ready signal.

Registers Affected: A Timing: 3 + N + wait (high-speed) and 2, 3 + 2N + wait

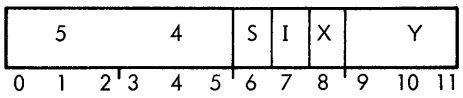
PIN PARALLEL INPUT



Pin stores the contents of 12 input lines in parallel in the effective memory location.

Registers Affected: M Timing: 5 + wait (high-speed) and 5, 6 + wait

BPI BLOCK PARALLEL INPUT



Starting with the effective memory location, BPI transfers N words from the 12 input lines in parallel into sequential locations. The contents of the A Register are the word count N minus one; BPI can input up to 4096 words per execution. The input lines fill and empty under the control of the Ready signal and BPI.

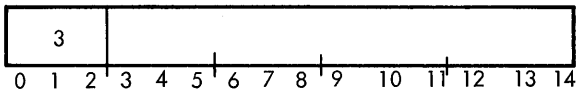
Registers Affected: M to M + (A), A Timing: 4 + N + wait (high-speed) and 3, 4 + 2N + wait

SINGLE BIT INPUT/OUTPUT

Operating in the System mode, the two instructions, ENERGIZE OUTPUT M (EOM) and SENSE EXTERNAL SIGNAL (SES), provide single-bit input/output transmissions.

Execution of an EOM (System Mode) causes a 1.15-microsecond signal to be transmitted to one of a possible 4096 signal destinations. The system EOM format is:

SYSTEM MODE EOM (effective address)

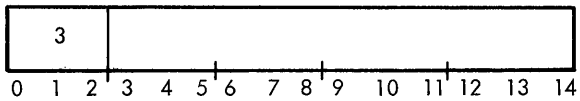


Bit positions 0-2 contain the System Mode Indicator.

Bit positions 3 through 14 contain the address field that specifies the special system destinations.

Registers Affected: None Timing: 3, 4

SYSTEM MODE SES (effective address)



Execution of an SES (System Test Mode) causes an address to be presented to the collection of special system devices. If the addressed external device is supplying a set signal to the central processor, the Flag Bit is set. If there is no signal, the Flag Bit is reset.

The SES System Test Format is identical to the System EOM Format. Timing is 3, 4 cycles.

I/O TERMINATION PROGRAMMING NOTES

1. There is a test to see whether the I/O channel is ready to accept a new input/output instruction – the CHANNEL ACTIVE TEST (CAT).
2. Following the termination of an input operation by an End-of-Record, the channel remains active until all significant data words have been stored by the program or interlace.
3. Following the termination of a non-magnetic tape output operation by TERMINATE OUTPUT (TOP), the channel remains active until the last character has been delivered to the peripheral device.
4. Following the termination of a magnetic tape output operation by TOP, the channel remains active until the magnetic tape unit commences stopping. This is long after the last character has been delivered to the magnetic tape unit.
5. The End-of-Record (I2) Interrupt is never sent until the I/O channel becomes inactive.

Thus, an input program (using End-of-Word, I1, Interrupts for example) must take care to store every input character presented to it by the I/O channel (in our example, every I1 Interrupt must result in a "WIN" or "DSC" or "TIP"). If not, the input device will proceed to End-of-Record and stop, but no I2 Interrupt will ever be given and the channel will never go inactive.

The programmer has several options after he has read as much of a record desired:

1. Disconnect (DSC) – but not if the peripheral device is a magnetic tape.
2. Do nothing – but only if no more words remain in the record (i.e., the entire record has been read).
3. Give TIP – the input peripheral device will continue to End-of-Record with all normal error checks on the remainder of the record.

4. TERMINATE INPUT (TIP) and TERMINATE OUTPUT (TOP) have the same octal configuration – the I/O channel differentiates TIP and TOP according to the type of operation it is performing.
5. The programmer can have no problem by giving a TIP when the input device is concurrently sending an End-of-Record signal or when the channel is already inactive.
6. Improper programming (especially on input) can leave the I/O channel in an active state.

THE I1 INTERRUPT

A non-interlace, non-character-interrupt I/O program should disarm the I1 Interrupt. If this disarming is not effected, spurious I1 Interrupts may result:

1. On output, the first I1 Interrupt is generated immediately following the activating EOM.
2. During ROT/RIN (and WOT/WIN), I1 Interrupts may be generated – even though the I/O channel is being properly attended to by the ROT/RIN command.

This disarming does not create many programming problems or compatibility problems – the RESET button clears both I/O channel interrupt arms. Thus, only programs which use both the interrupt and the non-interrupt modes of I/O programming need take heed.

PRIORITY INTERRUPT SYSTEM (Optional)

INTRODUCTION

As an option, the SDS 92 may contain a priority interrupt system. This system provides added program control of input/output operations, aids in programming multiplexed operations, and allows immediate recognition of special external conditions.

When an interrupt is received, the internal logic examines the interrupt signal and causes the computer to interrupt the program sequence at the end of the execution cycle of the current instruction. Without disturbing the Program Counter Register, the computer transfers program control to one of a selected set of memory locations. One of the branch and mark place BRM or BMC instructions in this location saves the contents of the Program Counter, Flag, and PCT. It also transfers to the particular interrupt servicing routine required. This enters the proper service routine since each interrupt has a unique interrupt location. To exit from the routine, a BRANCH AND CLEAR INTERRUPT (BRC) instruction using indirect addressing returns control to the next instruction in proper sequence in the main program; it also clears the interrupt and restores the original contents of the Flag and PCT.

The priority interrupt system has up to 256 System interrupt levels. The levels, numbered evenly upward from 200_8, have priority according to number, with the higher priority levels having a smaller number.

Additional interrupts obtained with SDS optional hardware are located at interrupt levels numbered from 150_8. In general, these also have priority according to number. Note that interrupts 150 through 176 have priority over any System interrupt (200 or more).

When an interrupt has occurred and its service subroutine has been entered, an interrupt of higher priority can interrupt the subroutine and gain program control for the servicing of its more important operation. But an interrupt of lower priority cannot interrupt an interrupt-processing subroutine of a higher level. Thus, the priority interrupt system allows interrupts to be arranged according to their importance and/or according to their need for speedy servicing.

The above type of interrupt is called a normal priority interrupt to differentiate it from another interrupt feature, the single-instruction interrupt. This is a different kind of interrupt that causes the execution of only one instruction before automatically clearing itself and returning to the program which it interrupted. If the executed instruction is a branch instruction which branches (i.e., BRM or BMC), the interrupt is cleared but control does not return to the interrupted routine. This type of interrupt needs no branch instruction to clear it. For example, by connecting an external clock source to the computer, the program can maintain a programmed real-time clock. Each time the external pulse causes an interrupt, the program executes the single instruction, MEMORY PLUS ONE TO MEMORY, to add one to the selected memory word. The main program examines this location whenever necessary to determine how many time increments have elapsed since the clock was started. No new interrupt can occur between any single-instruction interrupt and the return to the main program.

Any of the optional, system interrupts can be single- or normal-instruction interrupts in any combination desired.

PRIORITY INTERRUPT OPERATIONS

A normal priority interrupt level has three operational states: Inactive, Waiting, and Active.

In the inactive state, no interrupt signal has been received into the level and none is currently being processed by its interrupt servicing subroutine. No record is maintained if the interrupt cannot go into the waiting state.

In the waiting state, an interrupt request signal has been received into the level, but is not being processed. This situation may be due to an interrupt of higher priority being processed at this time. When the system is enabled and all higher waiting interrupts have been processed, this level goes to the active state.

In the active state, the interrupt has been acknowledged, meaning it has caused the main program to recognize its presence and has transferred to its assigned interrupt location and/or routine where it is being processed. When the interrupt processing is completed, execution of a BRANCH AND CLEAR (BRC) sets the interrupt level to the inactive state.

A single-instruction interrupt operates in the same way as the normal priority interrupt in the inactive and waiting states. However, when acknowledged, this interrupt enters the active

state, and remains there during the execution of one instruction. At the completion of the one instruction, the single-instruction interrupt returns to the inactive state without the aid of a branch and clear instruction.

INTERRUPT CONTROL

Two program control features are available in the interrupt system. These features are Enable/Disable and Arm/Disarm. Arm/Disarm (optional hardware) controls whether an interrupt can proceed from the inactive state to the waiting state. The disarm condition of an interrupt level prohibits an interrupt signal entering the level from causing the interrupt to enter "waiting" from "inactive".

With Enable/Disable, the entire set of interrupts in the system can be enabled and disabled under program control. When the interrupt system is enabled, interrupts can proceed from the waiting state to the active state.

The following interrupts are exceptions and are always enabled:

1. Power fail-safe (2 interrupts)
2. Memory parity error (2 interrupts)
3. Real-time clock (2 interrupts)
4. I/O Channel (2 interrupts)

The control of the optional Arm/Disarm feature operates on individual interrupt levels of the System interrupts (200-1176), that is, any chosen interrupt level may be selectively armed or disarmed. But the instruction structure for Arm/Disarm allows these interrupts to be operated on in groups of sixteen.

SINGLE INSTRUCTION INTERRUPTS ROUTINES

Only the following instructions will be meaningfully interpreted as single-instruction interrupt routines:

1. EOM
2. BMC, BRM
3. MPO – MPO, in this case, will not alter the Flag. However, if the restored, incremented operand equals 0000$_8$, a different interrupt pulse will be generated (see Real-Time Clock Option, Appendix B-1).
4. EXU – Can only execute the above-listed instructions.

NON-INTERRUPTABLE INSTRUCTIONS

An interrupt cannot occur between the execution of ENERGIZE OUTPUT M (EOM) and the instruction following it. This is also true for the input/output instructions, POT/BPO, PIN/BPI, WOT/ROT, and WIN/RIN. No interrupt can occur between a single-instruction interrupt and the return to the main program. When these instructions branch, an interrupt cannot occur between their execution and the execution of the branch-to instruction:

BRU	BDA	BFF
BRL	BAX	BFT

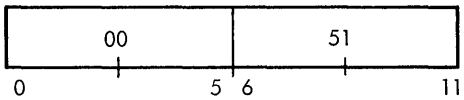
Table 3-2. Interrupt Location Assignments

150	Power On (always armed)
152	Power Off (always armed)
154	Main Frame Parity Error (armed via a console switch)
156	Data Multiplexing System Parity Error (armed via a console switch)
160	Unassigned
162	Unassigned
164	Interrupt, Clock Sync (always armed)
166	Interrupt, Clock Pulse (arm furnished)
170	I1 (arm furnished)
172	I2 (arm furnished)
174	Unassigned
176	Unassigned
200	System Interrupts (arms optional, single instruction discretionary)
1177	System Interrupts

ENABLE/DISABLE INTERRUPT INSTRUCTIONS

Three instructions are available for setting, resetting, and testing the state of the INTERRUPT ENABLED indicator.

EIR ENABLE INTERRUPT

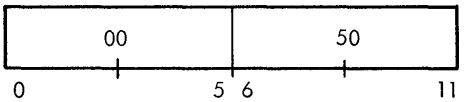


EIR unconditionally sets the INTERRUPT ENABLED indicator and enables the interrupt system. At the end of the next interruptable instruction, if any interrupt levels are waiting, the one with the highest priority becomes active.

EIR cannot be interrupted.

Registers Affected: None Timing: 3, 4

DIR DISABLE INTERRUPT

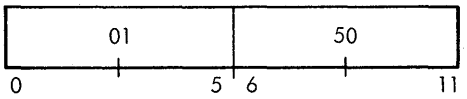


DIR unconditionally resets the INTERRUPT ENABLED indicator and disables the interrupt system. The current state of all interrupt levels is unchanged by this instruction.

DIR cannot be interrupted.

Registers Affected: None Timing: 3, 4

IET INTERRUPT ENABLED TEST; SET FLAG IF INTERRUPT SYSTEM ENABLED



If the priority interrupt system is enabled, IET sets the Flag Bit. If the priority interrupt system is disabled, IET resets the Flag Bit.

Registers Affected: F Timing: 3, 4

ARMING FEATURE

The arming feature is controlled for a group of 16 interrupts at one time. (The 24-bit POT/PIN option is a prerequisite for the arming feature.)

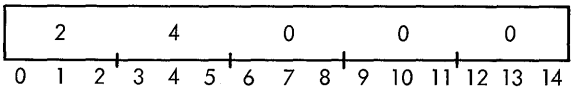
The sequence of instructions required to arm the selected interrupts is:

EOM		Alert the extender
POT	HITWELVE	Load most significant 12 bits into extender
AIR		Arm interrupts
POT	LOWTWELVE	Transmit entire 24 bits to the arming chassis and arms selected interrupts

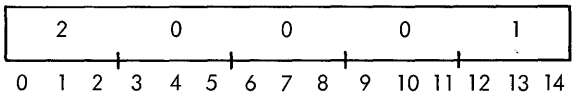
EXTENDER ALERT EOM

The Extender EOM alerts the extender to accept the next "POTted" word. AIR alerts the arming chassis that a 24-bit control word is coming with the POT that follows. The second POT triggers the entire 24-bit transmission.

The effective address of the Extender Alert EOM (an internal type) is:



AIR ARM INTERRUPTS



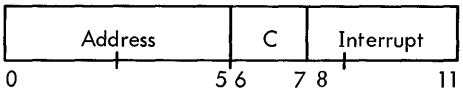
AIR prepares the arm interrupt control unit to receive a control word for a group of 16 interrupt levels. A PARALLEL OUTPUT (POT) must always follow AIR, or an unpredictable operation results.

AIR cannot be interrupted.

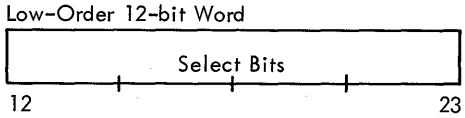
Registers Affected: None Timing: 3,4

The two words which the PARALLEL OUTPUT (POT) instructions address have the following format:

High-Order 12-bit Word



Low-Order 12-bit Word



The address field in bit positions 0 through 5 identifies which group of 16 interrupts in the system is being addressed. The C field controls what is done to the particular interrupt levels selected in bit positions 8 through 23. Bit position 8 refers to the lowest-numbered level of the group, therefore the one with the highest priority. Bit position 23 refers to the last or highest-numbered level, the one with lowest priority. The first group of 16 is group 0. For example, words of 0024 and 0000 arm level number 202 (the level of second highest priority).

The control operations are:

6-7	Function
00	Not used
01	Arm all interrupt levels that are selected by a 1 in bit positions 8-23
10	Disarm all interrupt levels that are selected by a 0 in bit positions 8-23
11	Arm all interrupts selected by a 1 and disarm all interrupts selected by a 0 in bit positions 8-23

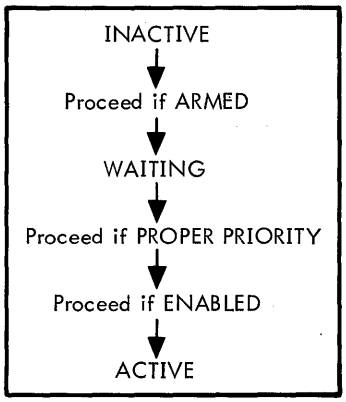


Figure 3-3. Interrupt Arm-Enable Response

CONTROL CONSOLE

The basic computer system includes a console for operator control. This console connects directly to the central processor, contains switches for operation, and displays the contents of operational registers.

DISPLAYS

The registers displayed on the console directly reflect the contents of the hardware registers. If the operator clears or changes a display, the contents of the actual register change identically.

PROGRAM LOCATION

This display consists of 15 binary indicators with a CLEAR button for the entire register and a set button for each indicator. The program counter contains the location of the next instruction to be executed. The operator may change the contents of the program counter via the CLEAR and set buttons. When the operator places the computer in RUN, the first instruction comes from the location shown in the PROGRAM LOCATION display.

INPUT-OUTPUT

The UNIT lights contain the unit address of the peripheral device currently connected to the I/O Channel.

The ERROR light reflects the status of the I/O Channel error indicator.

HALT

The HALT light lights whenever the computer executes a halt instruction while in the RUN position. Setting the RUN/IDLE/STEP switch to IDLE clears the halt.

REGISTER DISPLAY

This display consists of 12 binary indicators with a CLEAR button for the entire register and a set button under the P Register indicators that also serve the A, B, and C Registers. The Register switch selects the internal register to be displayed. The selectable registers are:

C C Register, which usually contains the contents of the memory word whose address is in the program counter

A A Register

B B Register

Placing the computer in IDLE, clearing the register, and then pressing the button in the corresponding bit positions under the indicators sets the contents of the selected register. Pressing a button places a 1-bit into the selected position of the register.

MEMORY PARITY

If an operand or instruction access from memory encounters a parity error and the memory switch is in the HALT position MEMORY PARITY lights. Setting the memory parity switch to CONTINUE clears the indicator and turns off the light.

ENABLE

ENABLE lights whenever the interrupt system is enabled.

FLAG

The FLAG indicator consists of a single binary indicator. FLAG is lit when the Flag Bit is a 1.

SWITCHES

POWER

The POWER switch turns the computer system power on or off. When power is on, the switch is lit.

FILL

To initiate a "fill",

1. Press the RESET button.
2. Hold down the FILL switch corresponding to the peripheral device from which a fill is desired.
3. Move the RUN/IDLE/STEP switch from IDLE to RUN while continuing to hold the appropriate FILL.
4. Release the FILL switch.

Fill causes the following:

1. An EOM opcode is generated.

   Bit 0 = 0

   Bit 1 = 0, Bit 2 = 0 ⟶ Buffer Control Mode

   Bit 3 = 0 ⟶ Forward Direction

   Bit 4 = 0

   Bit 5 = 1 ⟶ Binary Mode

   Bit 6 = 0

   Bit 7 = 1 ⟶ Two Characters per Word

   Bit 8 = 1

   Bit 9 - Bit 14 = The unit address of the indicated peripheral:

   Paper Tape = 04_8

   Cards = 06_8

   Mag Tape = 10_8

   Disc = 26_8

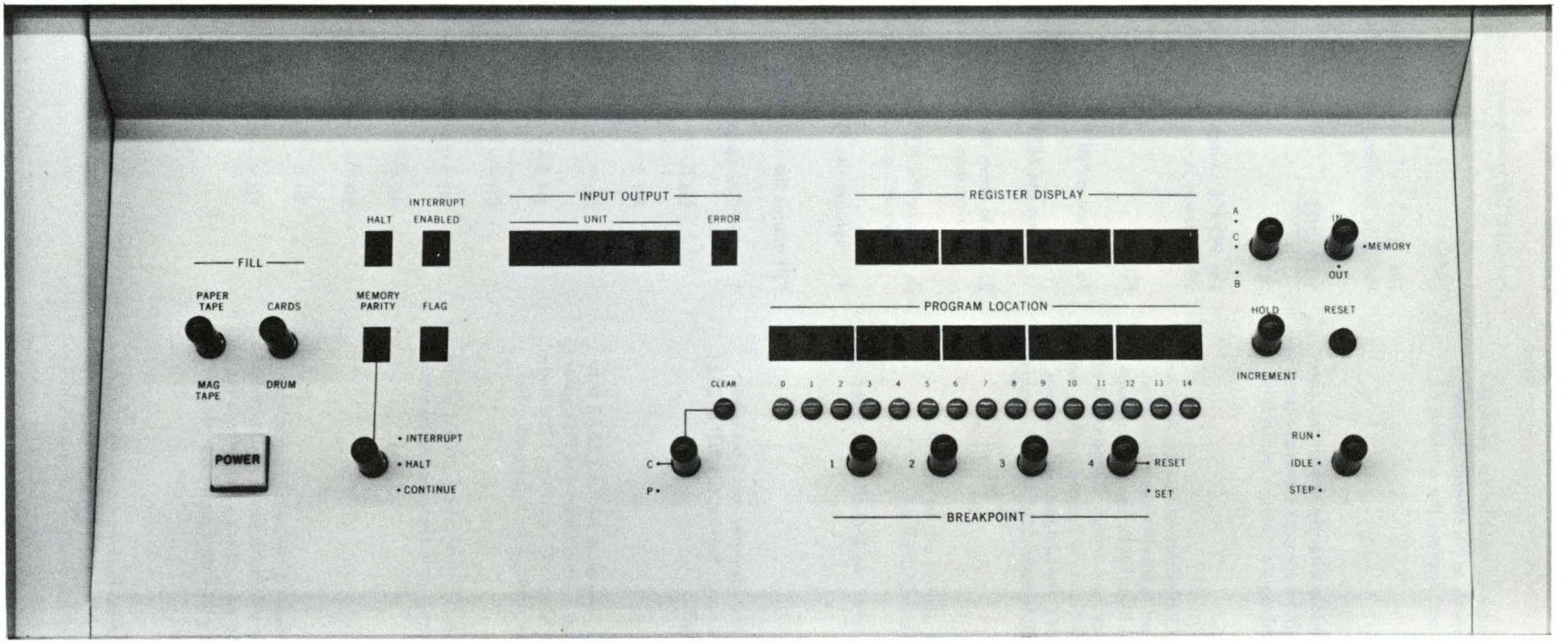


Figure 3-4. SDS 92 Computer Control Panel

2. 9 words (or to End-of-Record) read into memory starting in location 00000_8.

3. Computation begins at location 00000_8.

4. NOTE: The I/O Channel is still active and the input peripheral device is still sending characters to the channel. (If the fill routine is less than 9 words, the End-of-Record makes the channel inactive; computation still goes to 00000.)

RUN/IDLE/STEP

The RUN/IDLE/STEP switch is a three-position, toggle switch with two stationary positions and a spring-loaded momentary position in STEP. In the RUN position, computation occurs at machine speed. In the IDLE position, the computer idles immediately after an instruction has been read from memory. If the Register switch is in the C position, the first word of an instruction may be viewed in the REGISTER DISPLAY. Depressing the switch to STEP reaccesses and executes the instruction; the computer returns to the Idle state. To "step" another instruction, the operator releases the switch to the IDLE position and then depresses it again to STEP. No interrupts can occur (i.e., go into the active state) while stepping.

HOLD/INCREMENT

Placing the HOLD/INCREMENT switch in the "up" position causes the current contents of the program counter to be held. This inhibits the program counter from counting.

Momentarily placing the HOLD/INCREMENT switch in the INCREMENT position increments the program counter by one and brings the contents of the newly addressed location to the C Register.

RESET

This switch initializes the control section of the computer. It resets the I/O Channel, clears the FLAG, sets PCT, clears the INTERRUPT ENABLED, clears any parity error indication, clears all interrupts arms, and clears all interrupt levels. The operator must set the RUN/IDLE/STEP switch to IDLE before pressing this switch.

Switch Select

This two-position toggle (labeled C and P) selects which REGISTER DISPLAY/PROGRAM LOCATION will be affected by the CLEAR and set buttons.

Register Select

This three-position switch selects the register to be shown on the REGISTER DISPLAY lights.

MEMORY PARITY

In the HALT position, this switch causes the computer to enter the Idle state whenever a memory parity error occurs. In the CONTINUE position, the computer does not change state when a memory parity occurs. In the INTERRUPT state, any memory parity will result in one of two interrupts (optional).

BREAKPOINT

The four BREAKPOINT switches are externally controlled, internally testable program switches. Breakpoint test instructions test them.

MEMORY OUT

This is a momentary switch that causes the computer, in IDLE, to place the contents of the location specified by the program counter into the C Register.

MEMORY IN

This is a momentary switch that causes the computer, in IDLE, to place into the location specified in the program counter the contents of the C Register.

PERIPHERAL EQUIPMENT DESCRIPTION

This section describes some of the input/output devices that can be attached to a buffer and explains their use.

INPUT/OUTPUT TYPEWRITER

The control console may contain an electric, input/output typewriter for operator control, error or status messages, and similar functions. The typewriter connects to the I/O Channel, has the input unit address 01, and the output unit address 41. Appendix I-1 contains the typewriter codes.

The typewriter control instructions follow. The sample instructions use Typewriter No. 1 with 2 characters per word mode.

RKB 1,2	READ KEYBOARD 2 Characters/Word	02301†

This instruction activates the I/O Channel and connects to it Typewriter No. 1. RKB readies the channel to read input from the keyboard. This instruction lights the input light on the typwriter.

TYP 1,2	WRITE TYPEWRITER 2 Characters/Word	02341

This instruction activates the I/O Channel and connects to it Typewriter No. 1. TYP readies the channel to write output to the typewriter.

† This octal number is the EOM or SES effective address configuration.

EXAMPLE: Typewriter Output

This example types the following message: DO

from location OUTWD; the internal codes for these characters are in this location. The routine uses Typewriter No. 1; the routine assumes the channel to be initially inactive.

Location	Instruction	Address	Comments
IN	RES	2	This assembler directive reserves two locations for the mark entry.
	TYP	1,2	This EOM instruction connects Typewriter No. 1 to the channel for output and specifies two characters per word. The octal configuration of the EOM address is 02341.
	WOT	OUTWD	This instruction transfers the contents of location OUTWD to the I/O Channel. The new contents of the channel are output to the typewriter as two 6-bit characters and typed. The next instruction in sequence is executed as soon as the word is placed in the channel.
	TOP		This instruction terminates output on the channel. When the channel and the Single Character Register are clear of characters to be output, the channel sets its Unit Address Register to zero; this disconnects the channel. When accessed, this instruction executes immediately; the next instruction in sequence is then executed. The octal configuration of this EOM address is 12100.
	BRU	*IN	This instruction transfers to some other program area.
OUTWD	24 46		This word contains the internal code for the characters DO.

EXAMPLE: Typewriter Output then Input

This example types out the message: PROG

then awaits the input of a single character. Input terminates with a carriage return typed by the operator; the housekeeping necessary to determine when the carriage return has been input is not given.

IN	RES	2	
	TYP	1,2	Connect channel to Typewriter No. 1.
	WOT	MSSGE	Output first word of message.
	WOT	MSSGE+1	The central processor "hangs up" on this instruction until the second character from the preceding instruction has cleared the channel buffer into the Single Character Buffer for output. Then this WOT executes filling the channel buffer with contents of location MSSGE+1.
	TOP		Terminate output when channel system is clear.
TEST	CAT BFF	 TEST	The program "hangs up" here until the channel transmits the last character.
	RKB	1,1	This instruction connects Typewriter No. 1 to the channel for input and specifies one character per word. The octal configuration is 02101.
	WIN	KEYWD	The computer "hangs up" on this instruction until a character enters the channel from the keyboard; then the word in the channel buffer is placed into location KEYWD. The input character is in bit positions 0 through 5 of KEYWD. Bit position 6 through 11 are unpredictable

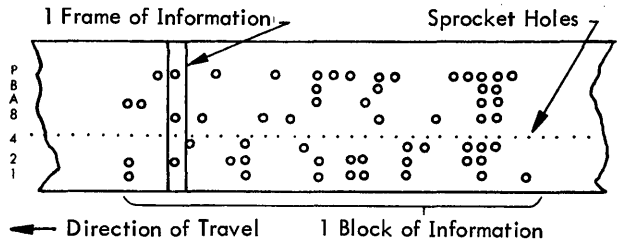
At this point, the word in KEYWD is placed elsewhere in memory and the routine returns to the WIN above. When executed, a test is made to determine if the new input character is the carriage return code. Indexing or indirect addressing can be used with the WIN to facilitate input. When the carriage return is detected, the following is executed.

	DSC		This instruction disconnects the channel by immediately clearing the Unit Address register to zero. The octal configuration of the EOM address is 00100.
	BRU	*IN	Return to main program.

PAPER TAPE INPUT/OUTPUT

Format

The paper tape used is one-inch wide, affording space for eight data holes and a sprocket hole in each frame of information. There are ten frames per inch of paper tape. Six hole positions are used for information, one is used for an odd parity check, and the eighth is unused.



Information is organized on the tape in blocks. A block is a group of frames set off by a gap of at least one blank frame (in which only the sprocket hole is punched) at either end. Blocks may be of variable lengths.

In some operations, a tape may consist of only one block, such as a source language tape prepared off-line. In this case, the program need not read the entire block at one time, but may stop the reader between frames, by disconnecting via DSC, and then start again to read the remainder or another portion of the block.

Reading

A program reads paper tape in a straightforward way, using RIN or a WIN in a read loop until the desired number of words are input or until gap is detected. The tape stops in less than one frame; this means that no frame is missed between subsequent read operations. An input operation that terminates because of gap (End-of-Record) stops the tape after the first blank of the gap. When starting the tape for reading, the tape reader ignores any leading blank frames. After reading a meaningful data word (one or two characters as defined by the program) from the tape, the reader recognizes the next blank frame as gap and signals the channel with an End-of-Transmission indication.

Punching

The EOM to alert the tape punch also turns on the punch motor (if not already on). If the punch instruction (EOM) so indicates, the punch unit punches a segment of leader (gap, or blank frames). Bit position 4 of the EOM that addresses the punch contains a "0" to punch leader; bit position 4 contains a "1" to punch without leader.

The EOM instruction that addresses and alerts the punch produces gap. No terminal punch operation produces gap after punching a block.

The punch operates asynchronously. If the channel does not supply characters to the punch fast enough, the punch waits for each character, losing no data and creating no errors.

Programming

There are no status tests for the reader or punch, that is, they are always ready for operation. When the channel addresses either device, the device starts to send or accept data within approximately one character time. The reader and punch operate only in the binary mode and the forward direction. The reader or punch ignores any different mode specified, and uses the forward-binary mode. Unit address of 04 is for Paper Tape Reader 1, and unit address 44 is for Paper Tape Punch 1.

Paper Tape Instructions

The following instructions use the I/O channel, Paper Tape Number 1 with two characters per word format.

RPT 1, 2 READ PAPER TAPE 02304

This instruction initiates a paper tape read operation on tape read station number 1 connected to the channel in the two characters per word format.

PTL 1, 2 PUNCH PAPER TAPE WITH LEADER 00344

This instruction initiates a paper tape punch operation on tape punch station number 1 connected to the channel in the two characters per word format. It generates approximately twelve (12) frames of leader preceding the first punched frame.

PPT 1, 2 PUNCH PAPER TAPE WITHOUT LEADER 02344

This instruction initiates a paper tape punch operation on tape punch station number 1 connected to the channel in the two characters per word format. It generates no leader preceding the first punched frame.

EXAMPLE: Punch Paper Tape

This routine punches one block of eight words (16 characters) from locations 02000 through 02007. A twelve-frame leader precedes the block. The routine is a closed subroutine.

Location	Instruction	Address	Comments
FRST	RES	2	This instruction is an assembler mnemonic used for convenience to reserve the subroutine entry locations.
	PTL	1,2	This instruction connects the channel to Paper Tape Punch No. 1 and specifies two characters per word mode. The instruction asks for leader to be punched. The octal configuration for this EOM is 00344.
	LDA	=7	This instruction sets (A) equal to 7.
	ROT	02000	This instruction transfers each word as needed to the channel beginning in location 02000.
	TOP		This instruction is executed in 4 or 5 cycles and then the computer executes the next instruction. The execution of TOP causes the channel to disconnect when the last character shifts out of the buffer and transmits out of the Single Character Register.
	BRU	*FRST	This instruction returns to the main program.

EXAMPLE: Read Paper Tape

This routine reads a 64-character block from paper tape into memory beginning at location 02000. The routine uses the two character per word mode, making the input 32 words. The routine is a closed subroutine.

FRST	RES	2	This assembler instruction reserves the entry locations.
	RPT	1,2	This instruction connects to the channel the Paper Tape Reader No. 1 and specifies two characters per word mode. The octal configuration for this EOM is 02304.
	LDA	=33	The 33 represents two more than the expected record size.
	RIN	02000	This instruction receives each word in the block beginning in location 02000 and going through 02037_8.

When the channel detects the End signal (gap) following the block during the input transmission, the RIN finishes execution and the computer goes to the next instruction.

CARD INPUT/OUTPUT

Format

The computer uses 80-column cards in two formats: Hollerith and binary. The reader reads Hollerith-coded information from cards and the corresponding SDS character codes go into memory. In this mode, each card column contains the equivalent of one 6-bit internal character. The character codes are in Appendix I-1.

Binary-coded information goes onto the card with two 6-bit characters per column. In binary mode, one column forms a word. The reader reads the card from column 1 to 80 in a top-bottom order. A single card holds 160 characters.

Figure 3-5 shows the relation of Hollerith information on a card and in memory.

Reading

The card reader scans the card, column by column, starting with column one, and transmits either 80 or 160 characters depending on the mode of operation. With power on and cards in the hopper, the operator readies the reader by pressing the START button. During program operation, the program must test for the Ready condition before initiating a card-read. A Read EOM instruction starts the card-reading operation; then the program controls the flow of information into memory via a RIN or WIN loop. The end of the card sets the End-of-Record condition. In the Hollerith mode, any column-read that is not punched in one of the 64 combinations listed in Appendix I-1 results in a Validity check. Presence of a Validity check causes an error signal to be sent to the channel and lights the VALIDITY CHECK light on the reader. If the stacker should become full, or the hopper empty, the reader goes Not Ready and lights the NOT READY light. The card reader remains in the Not Ready state until the operator corrects the situation and presses the START button. Upon reading the last card, the reader sets an End-of-File signal if its EOF ON switch is on. The central processor can test the End-of-File signal which determines if more cards are in the hopper.

Punching

The punch punches cards a row at a time, starting with row 12. The punch coupler in both Hollerith and binary modes automatically rearranges the information to be punched. The card punch program must present the entire image, 80 or 160 characters, to the punch 12 times for each card. The punch operates in the following manner. As each row of the card approaches the punch station, the coupler examines every character of the image to determine which column position in that row should be punched. After the 12th output, the punch punches row 9 and completes the card cycle.

The card punch is Ready to punch if there are cards in the magazine, the stacker is not full, and the START button has been

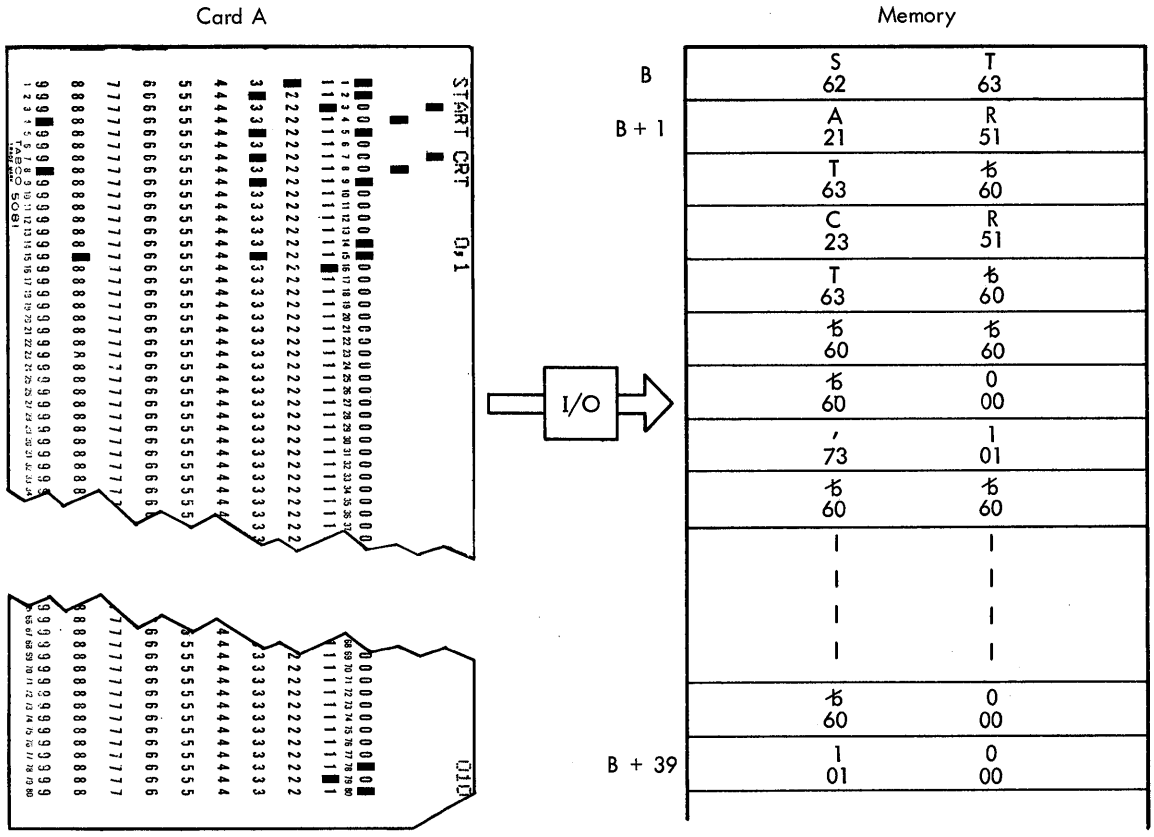


Figure 3-5. Card Read Into Memory in Hollerith

pressed. The punch remains Ready as long as the above conditions are true. A punch card instruction given when the punch is Ready causes a card to feed past the punch station. The program must then give the same instructions 12 times to transmit the card image to the coupler.

Programming Instructions

The card reader instructions below use unit number 1 with the two characters per word transmission mode.

CRT 1 CARD READER READY TEST 12106

This test determines if the selected card reader is Ready to read. If the reader is Not Ready, the computer resets the Flag Bit.

CFT 1 CARD READER END-OF-FILE TEST 11106

This test determines if the End-of-File condition from the card reader has been detected. If not, the computer sets the Flag Bit. If the EOF condition has been detected, the computer resets the Flag Bit.

The reader remains in the End-of-File condition until cards are added to the hopper or until the EOF ON switch is turned off.

RCD 1,2 READ CARD DECIMAL (Hollerith) 02306

RCD alerts the card reader, causes a card to feed from the hopper, and selects the Hollerith mode (as each column is read, it is translated to an SDS internal code). This mode reads up to 80 characters (40 words) from a card.

RCB 1,2 READ CARD BINARY 03306

RCB alerts the card reader, causes a card to feed from the hopper and selects the binary mode (as each column is read it is transmitted as two 6-bit binary characters). This mode reads up to 160 characters (80 words) from a card.

Card Punch Instructions

CPT 1 CARD PUNCH READY TEST 14146

This test determines if the selected card punch is Ready to punch. If so, the computer sets the Flag Bit. If the punch is Not Ready, the computer resets the Flag Bit.

The operator makes the punch Ready by placing blank cards in the magazine and pressing the START button.

PCD 1,2 PUNCH CARD DECIMAL (Hollerith) 02346

PCD alerts the punch, causes a card to feed past the punch station and selects the Hollerith mode. A transmission of 80 characters (40 words) must follow this instruction. The instruction PCD followed by the transmission instructions for 80 characters per card must be repeated 12 times.

PCB 1,2 PUNCH CARD BINARY 03346

PCB alerts the punch, causes a card to feed past the punch station and selects the binary mode. A transmission of 160 characters (80 words) must follow this instruction. The instruction PCB followed by the transmission instructions for 160 characters per card must be repeated 12 times.

EXAMPLE: Card Read

This program reads one card in Hollerith mode. It is a closed subroutine. The program enters the routine via a BRM.

Location	Instruction	Address	Comments
FRST	RES	2	This assembler instruction reserves locations for the subroutine entry.
TEST	CRT	1	This instruction is the card reader Ready test for Card Reader Number 1. It sets the Flag Bit if ready.
	BFF	TEST	This instruction branches back to the test on Not Ready. An exit to a Not Ready corrective routine can be put here.
	RCD	1, 2	This instruction connects the Card Reader 1 and starts a card moving toward the read station. Hollerith mode is specified. The octal configuration for this instruction is 02306.
	LDA	=39	This is the repeat count for RIN.
	RIN	READ	Beginning in READ, this instruction transfers words from the channel into the locations until the entire card is read.
	BRU	*FRST	This instruction branches back to the main program.

EXAMPLE: Card Punch

The program punches one card in Hollerith mode beginning from location 03740. The B Register counts the 12 times the program presents the card image to the punch.

FRST	RES	2	This instruction reserves the locations for the subroutine entry.
	LDB	=11	
TEST	CPT	1	This instruction tests the card punch for a Ready condition. It sets the Flag Bit if Ready.
	BFF	TEST	This instruction branches back to the test, CPT 1, if the Flag is reset. An exit to a time loop with the facility to tell the operator that the card punch will not become Ready can be placed here.
GEE	PCD	1, 2	This instruction executes if the punch is Ready. It connects the channel to the Card Punch Number 1, and starts a card moving toward the punch station. The two characters per word and Hollerith mode are specified.
	LDA	=39	Starting with location 03740, ROT transmits 40 words to the Punch.
	ROT	03740	
	TOP		This terminates output.
CTEST	CAT		Wait for the last character to be transmitted.
	BFF	CTEST	
	SUB	=1	SUB decrements (B) by one and sets F if the new (B) is equal to 7777_8.
	BFF	GEE	Note that the card image must be sent to the channel twelve times to punch a card.
	BRU	*FRST	Return to main program via location FRST.

MAGNETIC TAPE INPUT/OUTPUT

Format

Magnetic tape units used in SDS computer systems are IBM-compatible. The tape is one-half inch wide, Mylar base material, 1.5 mils thick. Tape reels (10 1/2 inch, plastic) contain up to 2400 feet of tape. A reflective marker, placed on the back of the tape approximately ten feet from the beginning of it, indicates the load point. The leading ten feet leave space for threading tape through the guides on the unit. The load point marker is on the Mylar side of the tape along the edge nearest the operator when the tape is mounted. A similar marker is along the other edge of the tape to mark the end-of-reel. About 14 feet of tape are reserved between the end-of-reel marker and the end of the tape. This space includes at least ten feet of leader and enough tape to hold a record of 9,600 characters in 200 bpi density after the end-of-reel marker is sensed.

Characters are recorded on tape in seven parallel tracks. A change in the magnetic flux in a track records a 1-bit for a given character position. No change in magnetic flux indicates a 0-bit. Six of the tracks contain information; the seventh track is a parity check. The system allows both even and odd parity, as needed. Binary recording uses odd parity. In this mode, the tape records the six-bit characters from memory without change. Binary-coded decimal (BCD) recording uses even parity. In this mode, the tape control unit transforms characters from the channel to conform with standard IBM, BCD interchange code (see Appendix A-1).

Only the capacity of available core storage in the computer limits block length. A record gap (section of blank tape) about 3/4-inch long separates blocks on tape. In writing, the tape automatically produces gap at the end of a record. Reading begins with the first character sensed after the gap and continues until the next gap is encountered.

An inter-record gap, followed by a special, single-character record, marks the end of a file of information. The character is a Tape Mark (0001111). Writing a one-word record in BCD with one-character-per-word format can record such a mark.

A program may write one or more files on a reel of tape. On reading an End-of -File record, the tape control unit stops the tape and sets its End-of-File indicator which may be tested by the program.

The tape control unit considers any record containing only Tape Mark (0001111) characters an End-of-File. The tape reads such characters into memory like any other characters.

As the tape unit writes information, it makes an odd-even count of the number of 1-bits in each track. At the end of each record, it writes a bit for each track such that the total number of 1-bits in each track is even. This parity check sum is always even whether the character parity is even or odd. The character containing these check bits is the longitudinal parity character; the tape unit writes it slightly past the end of recorded information in the block.

The longitudinal check character always reflects an even parity check for each channel. In the BCD mode, the check character itself always has an even number of 1-bits. In the binary mode, however, the check character may have either an even or an odd number of 1-bits. This means that a reverse scan over a binary record may result in turning on the error indicator in the buffer even though the record itself is correct. As a general rule, the program ignores the error indicator after a reverse operation.

Routines should always place a TAPE READY TEST (TRT) between tape operations of opposite direction to ensure that the tape unit stops and reverses. Good programming terminates tape writing by several inches of erasure whenever subsequent resumption of recording is anticipated. This eliminates the effects of a possible extraneous character that might arise through subsequent tape repositioning.

Reading

A Read Binary or Read BCD EOM starts a tape which continues until the tape unit detects an End-of-Record gap. If the computer does not instruct the tape unit to continue, it stops in the middle of that gap. When the tape stops, the tape unit disconnects from the channel. If the tape encounters an End-of-File, the tape control unit sets its EOF indicator. The central processor can test this indicator which remains set until the tape unit control receives a new EOM. The tape always stops after the Tape Mark.

At the end of the file, the program reads the EOF character (0001111) into memory along with its check character. In a two character per word read, this appears in the first word of the input area as a 1717 word.

When the tape unit is writing on tape, it may transmit flux disturbing surges ahead of the current writing position; these surges affect previously written records further down the tape. This means that a record in the middle of a file cannot be updated or rewritten if the records that follow it are to be read.

Any error detected either by the channel in the character parity check or by the control unit with the longitudinal parity check sets the channel error indicator. When detecting such an error in reading, the routine should backspace the tape over the erroneous record and attempt to "re-read" the record.

The tape backspaces over records using the Scan feature. A Scan reverse EOM starts the tape in reverse. A TERMINATE INPUT (TIP) EOM should immediately follow. The program then waits for the channel to become inactive (or awaits the End-of-Transmission interrupt if armed and the interrupt system is enabled). When the channel becomes inactive (or the End-of-Transmission interrupt occurs), the tape stops in front of the backwardly traversed record.

A Scan operation is similar to a Read operation except that the channel shifts the characters read through its Word Assembly Register, but does not consider a word complete until a tape

gap is encountered. When the tape reaches the gap, the channel uses the last two characters in the word assembly as the only word read from the record. When scanning in reverse, the word consists of the last two characters scanned that are the first two logical characters of the record. This operation assembles these characters in reverse. For example, if the first two characters of the record are 12 and the tape scans the record in reverse, these appear as 21 in the word stored for that record.

The same operations occurs in the forward scan with the last two characters of the record forming the word stored. The Scan is useful for reverse searching on the first word of the records in the file being searched. In this case, the routine starts the tape in a reverse scan and "hangs up" on a WIN. When the tape reaches the beginning of the record, the first word of the record transfers to the buffer. The WIN stores the first word and the program checks the key word against the search key. If they agree, then the program need only wait for the channel to become inactive and the routine reads the record forward. If the record is not the desired one, the program gives another Scan reverse without waiting for the channel to become inactive.

If the tape encounters the End-of-Reel marker while reading, the tape logic sets the End-of-Reel indicator in the tape unit; the program can test this at any time. An End-of-File normally indicates the end of recorded information on tape. Possibly, however, the End-of-Reel indicator may mark the last record on the reel.

Writing

A Write routine writes tape after testing the tape unit for Ready and testing for the file protect ring on the tape reel (i.e., the flag was set by the test). The Write tape EOM starts tape motion; the tape remains in motion until it receives the termination signal from the channel. The tape control unit then writes the remaining characters of the record (those in the channel buffer) and writes the longitudinal check character. When the read-after-write head reads this check character, the tape signals the channel that it has reached Gap. If the tape receives no further Write instruction within one millisecond, the tape stops and disconnects.

If the user wishes to backspace or rewind and then to return at some later time to record additional information at the end of the previous series of records, the routine should write an End-of-File character or erase a segment of tape after the series of written records. This practice provides positive identification of the end of a file and facilitates return to a specific location on the tape. If the programmer does not use this method, the tape may not subsequently stop in the same location at the end of the series of records as it did when writing the last record. This would leave a segment of tape in the gap which has not been written and may cause erroneous operation when reading the tape.

In addition to writing under program control, the program can also erase tape. When an Erase EOM with an erase unit address is used, the tape operates as though it were in a Write mode, except that it records no information. The program counts the number of words to be erased.

The use of this type of erase is for the correction of a Write error. When a Write error occurs, an ERASE REVERSE TAPE starts the tape in reverse. Then the same count used to write the record originally controls the erase. This procedure ensures that the tape always returns to the beginning of the erroneous record, even if a bad spot on the tape might appear as a gap. The routine may now rewrite the record. If the Write still produces an error, the routine erases the record backward and then erases it forward, using the same count and bypassing the section of tape where the difficulty occurred. The routine may now rewrite the record on a new section of tape.

The erase procedure is used to produce the required 3.75 inches of blank tape between the load point and the first record. A routine does this by erasing 300 words at 200 bpi density, 834 words at 556 bpi density, or 1200 words at 800 bpi density.

EOM instructions to the tape units specify start-without-leader since the tape unit generates gap at the end of all records for leader. A leader instruction should never be included in a magnetic tape program because an attempt to generate leader may cause an erroneous operation.

Programming

The SES and EOM instructions for normal tape operations are listed below. The EOM instructions use two characters per word format.

TRT n	TAPE READY TEST	1051n

TRT test tape unit number n for Not Ready. If the tape is Not Ready, the computer sets the Flag Bit. If the tape is Ready, the computer resets the Flag Bit.

A tape is Not Ready:

if there is no physical unit set to the logical unit number being tested,

if the selected unit is not in the Automatic mode, or

if the tape is in motion for any operation.

FPT n	FILE PROTECT TEST	1411n

FPT tests tape unit number n for file protect ring. If the file ring is inserted, the computer sets the Flag Bit. If not inserted, the computer resets the Flag Bit. The reset will occur if logical unit n is absent from the channel line.

BTT n BEGINNING OF TAPE TEST 1211n

BTT tests tape unit number n for the beginning of the tape. If it is not positioned on the load-point marker, the computer sets the Flag Bit. If positioned at the load-point marker, the computer resets the Flag Bit. The reset will occur if logical unit n is absent from the channel line.

ETT n END OF TAPE TEST 1111n

ETT tests whether tape unit number n is not positioned at the end of the tape. If the tape unit has not sensed the End-of-Reel marker, the computer sets the Flag Bit. If the End-of-Reel marker has been sensed, the computer resets the Flag Bit. The End-of-Reel condition is reset when the tape is moved backward over the End-of-Reel marker. The reset will occur if logical unit n is absent from the channel line.

DT2 n DENSITY TEST, 200 BPI 1631n

DT2 tests tape unit number n for being set at 200 bpi density. If not, the computer sets the Flag Bit. If so, the computer resets the Flag Bit.

DT5 n DENSITY TEST, 556 BPI 1671n

DT5 tests tape unit number n for being set at 556 bpi density. If not, the computer sets the Flag Bit. If so, the computer resets the Flag Bit.

DT8 n DENSITY TEST, 800 BPI 1731n

DT8 tests tape unit number n for being set at 800 bpi density. If not, the computer sets the Flag Bit. If so, the computer resets the Flag Bit.

TFT TAPE END-OF-FILE TEST 13710

TFT test the tape control unit for a tape under its control encountering an End-of-File during the last Read or Scan operation. If the End-of-File indicator is reset, the computer sets the Flag Bit. If the End-of-File indicator is set, the computer resets the Flag Bit.

The End-of-File indicator remains set until another tape operation is requested.

WTB n,2 WRITE TAPE IN BINARY 0335n

WTB starts tape unit n in a Binary Write mode.

WTD n,2 WRITE TAPE IN DECIMAL (BCD) 0235n

WTD starts tape unit Wn in a BCD Write mode.

EFT n,2 ERASE FORWARD TAPE 0337n

EFT starts tape unit n in an Erase mode.

ERT n,2 ERASE REVERSE TAPE 0737n

ERT starts tape unit n in reverse in an Erase mode.

RTB n,2 READ TAPE IN BINARY 0331n

RTB starts tape unit n in a Binary Read mode.

RTD n,2 READ TAPE IN DECIMAL (BCD) 0231n

RTD starts tape unit n in a BCD Read mode.

SFB n,2 SCAN FORWARD IN BINARY 0333n

SFB starts tape unit n forward in a Binary Scan mode.

SFD n,2 SCAN FORWARD IN DECIMAL (BCD) 0233n

SFD starts tape unit n forward in a BCD Scan mode.

SRB n,2 SCAN REVERSE IN BINARY 0733n

SRB starts tape unit n in reverse in a Binary Scan mode.

SRD n,2 SCAN REVERSE IN DECIMAL (BCD) 0633n

SRD starts tape unit n in reverse in a BCD Scan mode.

REW n REWIND 1411n

REW starts tape unit n in a Rewind.

RTS 0 CONVERT READ TO SCAN 14100

RTS converts an in-process Read operation to a Scan. If the interrupts are disabled when the gap is encountered and the program is hanging on a WIN (executed after RTS, but before the gap), the WIN brings into memory the last two characters from the channel buffer. If the interrupts are enabled, an End-of-Word (I1) interrupt occurs when the gap is encountered by the tape unit; the last character is available via a WIN. If another Read or Scan EOM is executed within 1 millisecond of the gap occurrence, the tape does not stop and no End-of-Record (I2) interrupt occurs; if not, an I2 interrupt occurs when the tape is actively stopping (1 millisecond).

Note: All scans must be in the 2 characters/word mode. This necessarily implies that the read operation preceding an "RTS" must have been in the 2 characters/word mode.

MAGNETIC TAPE EXAMPLE PROGRAMS

The following examples show samples of complete input/output programs for magnetic tape.

EXAMPLE: Magnetic Tape Read

This program reads one record from Magnetic Tape No. 1 on the I/O Channel. The program is a closed subroutine. The tape is not at the beginning or the end of the tape.

Location	Instruction	Address	Comments
FRST	RES	2	This instruction reserves locations for the subroutine entry.
TEST	TRT	1	This instruction tests Ready Magnetic Tape No. 1. The octal configuration for the command is 10511.
	BFT	TEST	This instruction branches back to TRT if the Flag is set. An exit to a routine that determines reason for the non-Ready condition can be placed here.
	RTD	1,2	This instruction activates the channel, connects it to Magnetic Tape No. 1, and starts tape motion. The two characters per word and BCD modes are specified.
	LDA	=99	This count is for the RIN instruction to read 100 words.
	RIN	03000	Read 100 words starting at location 03000.
	TIP		Terminate input.
	BRU	*FRST	This instruction branches back to the main program via FRST.

EXAMPLE: Write Magnetic Tape

This program writes one record on magnetic tape. The program is a closed subroutine; it uses Magnetic Tape No. 1.

FRST	RES	2	This instruction reserves locations for the subroutine entry.
TEST	TRT	1	This instruction tests whether Magnetic Tape No. 1 is ready.
	BFT	TEST	This tests the Flag True. This instruction branches back to the Ready test if the Flag is set.
	FPT	1	This instruction tests whether the file protect ring is present on the tape reel. If so, the computer sets the Flag Bit. The octal configuration of the address is 14111.
	BFF	BRML	If the Flag is reset, branch to BRML.
	WTD	1,2	This instruction connects the channel to Magnetic Tape No. 1, specifies BCD transfer mode, and starts the tape moving. Two characters per word mode is specified. The octal configuration of the instruction is 02351.
	LDA	=99	The 100 is the block length.
	ROT	02000	Starting at location 02000, ROT writes 100 words.
	TOP		This terminates output.
CTEST	CAT		Wait for channel to disconnect.
	BFF	CTEST	
	BRU	*FRST	This instruction branches back to the main program via FRST.
BRML	BRM	OPER	This instruction branches and marks to an assumed routine to call the operator and instructs him to insert file-protect ring on Magnetic Tape No. 1.

LINE PRINTER

SDS buffered line printers are capable of printing up to 1000 lines per minute at 132 characters per line, with a standard set of 56 characters. Printing is accomplished by means of a rotating character drum and a bank of 132 print hammers. The drum passes 56 different characters, in lines of 132 each, past the hammer bank. Upon command from the computer, the selected print hammers drive the paper against the ribbon and onto the appropriate character typeface as it passes the print position. The characters are transmitted sequentially for storage in the printer buffer before printing. A programmable format tape loop provides fixed (or preselected) space control. Upspacing of 1 to 7 lines, as well as page control, may be accomplished by program instructions.

An optional, off-line facility allows the program or the operator to initiate card-to-printer or magnetic tape-to-printer operations simultaneous with computation (see Off-Line Printing).

Printer Controls

The printer controls, Figure 3-6, for SDS line printers consist of eight switches and indicators.

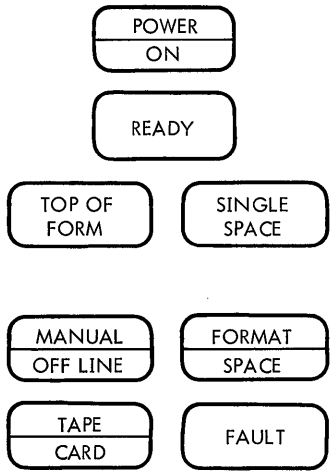


Figure 3-6. Printer Control Indicator Lights and Switches

The POWER ON switch is an alternate action switch. The computer must be turned on for this switch to be activated. Pressing POWER ON lights the top half of the indicator, turns on the motors and hammer driver power supply, and starts a timer that allows the motors to reach proper speed. After 20 seconds the bottom half lights, indicating that the printer is operable.

When the printer is initially turned on, the READY indicator is off. When pressed, it is turned on if:

1. paper is loaded in the line printer,
2. the lower half of the POWER ON switch is lighted, and
3. the hammer power supply is on.

This indicator automatically goes off when the above conditions are not realized. The printer is ready for either on-line or off-line operation when READY is turned on. Ready is reset to preclude computer intervention while changing paper or ribbon, or operating the TOP OF FORM or SINGLE SPACE switches.

Pressing TOP OF FORM causes the printer to position paper according to format tape channel 1. This indicator is lighted only when the format tape is positioned at channel 1, that is, top-of-form on a standard tape loop. This switch is operative when there is paper in the printer and the READY indicator is off.

Pressing SINGLE SPACE causes the printer to upspace paper one single space, independently of the vertical format tape. This switch is operative when there is paper in the machine and READY is off.

The FAULT indicator lights when the printer detects a parity error as information transfers from the buffer to the print hammers, or when it detects a parity error in incoming data from magnetic tape or cards during an off-line operation. It remains lighted until the next EOM addresses the printer. The condition of the light corresponds to the status of a program-testable fault indicator in the printer.

MANUAL OFF LINE† is a combination switch and indicator for off-line operation. The computer or the operator may initiate off-line operation, which is indicated by the illumination of the bottom half of this switch. If the operator presses this switch to initiate off-line operation, the top half is also lighted. This indicator is normally reset when the end-of-file is detected from the input unit. Pressing READY when it is lighted also resets it, that is, by switching the printer from the "ready" to the "not ready" state.

The FORMAT/SPACE† switch is used in off-line operation. The operator may use either mode, spacing a single space after each line of print, or using the first character stored on tape or cards as a vertical format character.

The TAPE/CARD† switch selects the desired input device.

Paper Tape Format Loop

A paper tape format loop, placed in the printer, allows upspacing to proceed to prespecified vertical positions on the print page. The format loop is an eight-channel paper tape. Putting a punch in the specified channel at the desired vertical spacing selects the channel upspace. Channel 1 is the top-of-form channel, channel 7 is the bottom-of-form channel, and channel 0 is the single-upspace channel. In the off-line mode with SPACE control, channel 0 controls single spacing. When printing with no format loop inserted in the printer, single upspacing occurs regardless of the channel specified.

Terminating Line Printer Output

When the single-word mode of transmission is used for printing on the line printer, each character transmission for a line must be followed by a TERMINATE OUTPUT (TOP) instruction. TOP is automatically generated with interlaced outputs.

Error Conditions

1. Print fault - parity error during transfer of character information from print buffer to print hammers.
2. Buffer error - parity or character rate error during transfer of information through buffer.

†If an off-line coupler is not attached to the printer, the MANUAL OFF LINE, FORMAT SPACE, and TAPE CARD indicators neither light nor affect printer operation.

3. Input fault - parity error in incoming data from cards or magnetic tape (during off-line operation only).

Off-Line Printing

The optional, off-line facility allows the line printer to produce printed records from card or magnetic tape sources without computer attention. The character transmission proceeds directly from the source to the computer for other input/output operations (e.g., card reading on card reader 2, card punch, paper tape read/punch, disk read/write, etc.). Once initiated, the printing operation is controlled by the source and proceeds until the source generates an end-of-file signal (see card input and magnetic tape input for appropriate end-of-file conditions).

The FAULT indicator lights when a parity error is detected during the reading of a tape record; the off-line printer rereads the record in an attempt to read good data. If this reread record contains an error, FAULT lights, the off-line operation terminates, and the printer goes back on-line if physically connected to the computer and the MANUAL indicator is off. When a validity check occurs during a card read, FAULT lights, the operation terminates, and the printer goes back on-line if the MANUAL indicator is off. The next EOM addressing the printer resets FAULT if the printer is on-line. If the MANUAL indicator is on, the error condition may be cleared by pressing READY off and then on again. If a fault occurs in an off-line operation initiated by the computer, the usual method for clearing the error is:

1. Press MANUAL on.
2. Press READY off.
3. Press READY on.
4. Press MANUAL off.

In a manually-initiated off-line operation, steps 1 and 4 are not required.

Off-line printing can be formatted as desired through the use of a single upspace or the format control mode (see Table 3-3). Off-line printing terminates by an end-of-file indicator from either device. Upon termination of an off-line operation, a physically connected off-line printer system returns on-line, provided the MANUAL indicator is off.

Table 3-3. Format Control Characters

Code	Character	Function
00	0	Skip to format channel 0
01	1	Skip to format channel 1
02	2	Skip to format channel 2
03	3	Skip to format channel 3
04	4	Skip to format channel 4
05	5	Skip to format channel 5
06	6	Skip to format channel 6
07	7	Skip to format channel 7
40	- (hyphen)	Do not space
41	J	Upspace 1 line
42	K	Upspace 2 lines
43	L	Upspace 3 lines
44	M	Upspace 4 lines
45	N	Upspace 5 lines
46	O	Upspace 6 lines
47	P	Upspace 7 lines

Printing Off-Line Under Operator Control

The procedure for operator control of off-line printing is:

1. Switch on the desired input device. (Magnetic tape is selected by dialing it to logical tape number 7.)
2. Place paper at top of form, as desired, by means of the TOP OF FORM switch.
3. Select desired input device by means of the TAPE/CARD switch.
4. Select either the FORMAT or SPACE mode as required.
5. Press MANUAL OFF LINE switch.
6. Press READY switch on, which initiates actual data transfer.

Printing Off-Line Under Computer Control

The procedure for computer control of off-line printing is:

1. Turn the equipment on.
2. Prepare the desired input device for operation.
3. Select desired input device by means of the TAPE/CARD switch.
4. Select either the FORMAT or SPACE mode as required.
5. Press the READY switch on.
6. Under program control, test the tape or card unit and the line printer for "ready" condition.
7. Then, to start transfer of data, give the POL instruction to print off-line.

Programming

SES and EOM instructions that have special use with the printer follow. For convenience, assume that the instructions address the channel and connect, test, or use Line Printer Number 1 on the channel.

PRT 1 PRINTER READY TEST 12160

This instruction tests the printer for a Ready condition. If the printer can accept a line to be printed, or accept a skip or space instruction, it is Ready. If the printer is Ready, the computer sets the Flag Bit. If the printer is Not Ready, the computer resets the Flag Bit.

When the printer is upspacing paper, PRT tests for Ready before the slew is complete. Therefore, PRT is ineffective for separating two successive upspace operations. The second upspace specified may override the first one unless sufficient delay is inserted (see PSP).

EPT 1 END OF PAGE TEST 14160

This instruction tests the printer for having paper positioned at the End-of-Page, which is marked by a punch in channel 7. If not at End-of-Page, the computer sets the Flag Bit. If at End-of-Page, the computer resets the Flag Bit.

PFT 1 PRINTER FAULT TEST 11160

This instruction tests whether the PRINT FAULT indicator is set. If not set, the computer sets the Flag Bit. If set, the computer resets the Flag Bit.

POL 1 PRINTER OFF-LINE 10360

This instruction places the printer off-line to begin an off-line print operation. The card reader and/or magnetic tape attached to the channel also goes off-line (see Off-Line Printing).

PSC 1, n PRINTER SKIP TO FORMAT CHANNEL n 1n560

The printer skips to format control channel n, where n denotes a channel number from 0 to 7. The format control is an eight-channel paper tape loop that is as long as the paper being used. (See PSP for timing.)

PSP 1, n PRINTER UPSPACE n LINES 1n760

The printer upspaces from 0 to 7 lines as specified by n. Consecutive upspace instructions must be separated by a sufficient time delay. Otherwise, the two PSP instructions may be merged by the printer.

Approximate completion times for PSP (from initiation of instruction to paper stop) are:

Upspace 1 line: 25 milliseconds
Upspace more than 1 line: Add 10 milliseconds for each additional line.

Off-Line Print Termination

Off-line printing terminates when an end-of-file indicator from the magnetic tape unit or card reader occurs. When printing from magnetic tape, the print operation terminates when the first character read from a record is the end-of-file code, octal 17.

When printing from cards, the print operation terminates when the end-of-file signal comes from the reader. This occurs when the card hopper becomes empty and the EOF ON switch on the reader is on (END OF FILE indicator lights). If the hopper becomes empty when EOF ON is not lighted, the printer waits for more cards to be placed in the hopper and the reader to become ready. When the reader is again ready, printing resumes.

EXAMPLE: Print Two Lines

This program prints two lines at the top of a page with a single upspace between. Assume that the printer is Ready or is becoming Ready after a print operation. The program is a closed subroutine for printer number 1.

Location	Instruction	Address	Comments
FRST	RES	2	Saves locations for subroutine entry.
	LDA	= 65	Load A with 65 for the length of a line image.
TST1	PRT	1	This instruction tests for printer Ready. If not Ready, the computer resets the Flag Bit. If Ready, the computer sets the Flag.
	BFF	TST1	Not Ready, return to the test.
	PSC	1, 1	This instructs the printer to move paper to the top of the page. The octal configuration is 11560.
	PLP	1, 2	Connect line printer to the channel, specify 2 character/word mode.
	ROT	LINE1	Output 66 words from line 1 image area.
	TOP		Terminate output.
TST2	CAT		Wait for channel to disconnect
	BFF	TST2	
	LDA	= 65	Reload A with 65.
TST3	PRT	1	Wait for printer to become ready after printing first line.
	BFF	TST3	
	PSP	1, 1	Upspace printer 1 line. The octal configuration is 11760.
	PLP	1, 2	Address printer.
	ROT	LINE2	Output image for line 2.
	TOP		Terminate.
	BRU	*FRST	Exit the subroutine via the BRU.

SDS CHARACTER CODES

Characters		SDS Internal Code	Card Code	Magnetic Tape BCD Code on Tape
Typewriter	Printer			
Ø	0	00	0	12
1	1	01	1	01
2	2	02	2	02
3	3	03	3	03
4	4	04	4	04
5	5	05	5	05
6	6	06	6	06
7	7	07	7	07
8	8	10	8	10
9	9	11	9	11
Space	Blank	12	8-2	12 (3)
# or =	=	13	8-3	13
@ or '	'	14	8-4	14
:	:	15	8-5	15
>	>	16	8-6	16
√	√	17	8-7	17
& or +	+	20	12	60
A	A	21	12-1	61
B	B	22	12-2	62
C	C	23	12-3	63
D	D	24	12-4	64
E	E	25	12-5	65
F	F	26	12-6	66
G	G	27	12-7	67
H	H	30	12-8	70
I	I	31	12-9	71
Backspace ? (1)	? (5)	32	12-0 (4)	72
.	.	33	12-8-3	73
¤ or)	)	34	12-8-4	74
[	[	35	12-8-5	75
<	<	36	12-8-6	76
‡ Stop	‡ (5)	37 (2)	12-8-7	77

Characters		SDS Internal Code	Card Code	Magnetic Tape BCD Code on Tape
Typewriter	Printer			
-	-	40	11	40
J	J	41	11-1	41
K	K	42	11-2	42
L	L	43	11-3	43
M	M	44	11-4	44
N	N	45	11-5	45
O	O	46	11-6	46
P	P	47	11-7	47
Q	Q	50	11-8	50
R	R	51	11-9	51
Car. Ret. ! (1)	! (5)	52	11-0 (4)	52
$	$	53	11-8-3	53
*	*	54	11-8-4	54
]	]	55	11-8-5	55
;	;	56	11-8-6	56
Δ	Δ	57	11-8-7	57
↑	Blank	60	Blank	20
/	/	61	0-1	21
S	S	62	0-2	22
T	T	63	0-3	23
U	U	64	0-4	24
V	V	65	0-5	25
W	W	66	0-6	26
X	X	67	0-7	27
Y	Y	70	0-8	30
Z	Z	71	0-9	31
Tab ‡ (1)	‡ (5)	72	0-8-2	32
,	,	73	0-8-3	33
% or (	(	74	0-8-4	34
←	← (5)	75	0-8-5	35
\	\	76	0-8-6	36
⁂ Delete	⁂ (5)	77 (2)	0-8-7	37

NOTES:

(1) The characters ? ! and ‡ are for input only. The functions Backspace, Carriage Return, or Tab always occur on output.

(2) On the off-line paper tape preparation unit, 37 serves as a stop code and 77 as a code delete.

(3) The internal code 12 is written on tape as a 12 in BCD. When read, this code is always converted to 00.

(4) The codes 12-0 and 11-0 are generated by the card punch; however, the card reader will also accept 12-8-2 for 32 and 11-8-2 for 52 to maintain compatibility with earlier systems.

(5) For the 64-character printers only.

TABLE OF POWERS OF TWO

2^n	n	2^{-n}
1	0	1.0
2	1	0.5
4	2	0.25
8	3	0.125
16	4	0.062 5
32	5	0.031 25
64	6	0.015 625
128	7	0.007 812 5
256	8	0.003 906 25
512	9	0.001 953 125
1 024	10	0.000 976 562 5
2 048	11	0.000 488 281 25
4 096	12	0.000 244 140 625
8 192	13	0.000 122 070 312 5
16 384	14	0.000 061 035 156 25
32 768	15	0.000 030 517 578 125
65 536	16	0.000 015 258 789 062 5
131 072	17	0.000 007 629 394 531 25
262 144	18	0.000 003 814 697 265 625
524 288	19	0.000 001 907 348 632 812 5
1 048 576	20	0.000 000 953 674 316 406 25
2 097 152	21	0.000 000 476 837 158 203 125
4 194 304	22	0.000 000 238 418 579 101 562 5
8 388 608	23	0.000 000 119 209 289 550 781 25
16 777 216	24	0.000 000 059 604 644 775 390 625
33 554 432	25	0.000 000 029 802 322 387 695 312 5
67 108 864	26	0.000 000 014 901 161 193 847 656 25
134 217 728	27	0.000 000 007 450 580 596 923 828 125
268 435 456	28	0.000 000 003 725 290 298 461 914 062 5
536 870 912	29	0.000 000 001 862 645 149 230 957 031 25
1 073 741 824	30	0.000 000 000 931 322 574 615 478 515 625
2 147 483 648	31	0.000 000 000 465 661 287 307 739 257 812 5
4 294 967 296	32	0.000 000 000 232 830 643 653 869 628 906 25
8 589 934 592	33	0.000 000 000 116 415 321 826 934 814 453 125
17 179 869 184	34	0.000 000 000 058 207 660 913 467 407 226 562 5
34 359 738 368	35	0.000 000 000 029 103 830 456 733 703 613 281 25
68 719 476 736	36	0.000 000 000 014 551 915 228 366 851 806 640 625
137 438 953 472	37	0.000 000 000 007 275 957 614 183 425 903 320 312 5
274 877 906 944	38	0.000 000 000 003 637 978 807 091 712 951 660 156 25
549 755 813 888	39	0.000 000 000 001 818 989 403 545 856 475 830 078 125
1 099 511 627 776	40	0.000 000 000 000 909 494 701 772 928 237 915 039 062 5
2 199 023 255 552	41	0.000 000 000 000 454 747 350 886 464 118 957 519 531 25
4 398 046 511 104	42	0.000 000 000 000 227 373 675 443 232 059 478 759 765 625
8 796 093 022 208	43	0.000 000 000 000 113 686 837 721 616 029 739 379 882 812 5
17 592 186 044 416	44	0.000 000 000 000 056 843 418 860 808 014 869 689 941 406 25
35 184 372 088 832	45	0.000 000 000 000 028 421 709 430 404 007 434 844 970 703 125
70 368 744 177 664	46	0.000 000 000 000 014 210 854 715 202 003 717 422 485 351 562 5
140 737 488 355 328	47	0.000 000 000 000 007 105 427 357 601 001 858 711 242 675 781 25
281 474 976 710 656	48	0.000 000 000 000 003 552 713 678 800 500 929 355 621 337 890 625

OCTAL-DECIMAL INTEGER CONVERSION TABLE

0000 to 0777 (Octal)	0000 to 0511 (Decimal)

Octal	Decimal
10000 -	4096
20000 -	8192
30000 -	12288
40000 -	16384
50000 -	20480
60000 -	24576
70000 -	28672

	0	1	2	3	4	5	6	7
0000	0000	0001	0002	0003	0004	0005	0006	0007
0010	0008	0009	0010	0011	0012	0013	0014	0015
0020	0016	0017	0018	0019	0020	0021	0022	0023
0030	0024	0025	0026	0027	0028	0029	0030	0031
0040	0032	0033	0034	0035	0036	0037	0038	0039
0050	0040	0041	0042	0043	0044	0045	0046	0047
0060	0048	0049	0050	0051	0052	0053	0054	0055
0070	0056	0057	0058	0059	0060	0061	0062	0063
0100	0064	0065	0066	0067	0068	0069	0070	0071
0110	0072	0073	0074	0075	0076	0077	0078	0079
0120	0080	0081	0082	0083	0084	0085	0086	0087
0130	0088	0089	0090	0091	0092	0093	0094	0095
0140	0096	0097	0098	0099	0100	0101	0102	0103
0150	0104	0105	0106	0107	0108	0109	0110	0111
0160	0112	0113	0114	0115	0116	0117	0118	0119
0170	0120	0121	0122	0123	0124	0125	0126	0127
0200	0128	0129	0130	0131	0132	0133	0134	0135
0210	0136	0137	0138	0139	0140	0141	0142	0143
0220	0144	0145	0146	0147	0148	0149	0150	0151
0230	0152	0153	0154	0155	0156	0157	0158	0159
0240	0160	0161	0162	0163	0164	0165	0166	0167
0250	0168	0169	0170	0171	0172	0173	0174	0175
0260	0176	0177	0178	0179	0180	0181	0182	0183
0270	0184	0185	0186	0187	0188	0189	0190	0191
0300	0192	0193	0194	0195	0196	0197	0198	0199
0310	0200	0201	0202	0203	0204	0205	0206	0207
0320	0208	0209	0210	0211	0212	0213	0214	0215
0330	0216	0217	0218	0219	0220	0221	0222	0223
0340	0224	0225	0226	0227	0228	0229	0230	0231
0350	0232	0233	0234	0235	0236	0237	0238	0239
0360	0240	0241	0242	0243	0244	0245	0246	0247
0370	0248	0249	0250	0251	0252	0253	0254	0255

	0	1	2	3	4	5	6	7
0400	0256	0257	0258	0259	0260	0261	0262	0263
0410	0264	0265	0266	0267	0268	0269	0270	0271
0420	0272	0273	0274	0275	0276	0277	0278	0279
0430	0280	0281	0282	0283	0284	0285	0286	0287
0440	0288	0289	0290	0291	0292	0293	0294	0295
0450	0296	0297	0298	0299	0300	0301	0302	0303
0460	0304	0305	0306	0307	0308	0309	0310	0311
0470	0312	0313	0314	0315	0316	0317	0318	0319
0500	0320	0321	0322	0323	0324	0325	0326	0327
0510	0328	0329	0330	0331	0332	0333	0334	0335
0520	0336	0337	0338	0339	0340	0341	0342	0343
0530	0344	0345	0346	0347	0348	0349	0350	0351
0540	0352	0353	0354	0355	0356	0357	0358	0359
0550	0360	0361	0362	0363	0364	0365	0366	0367
0560	0368	0369	0370	0371	0372	0373	0374	0375
0570	0376	0377	0378	0379	0380	0381	0382	0383
0600	0384	0385	0386	0387	0388	0389	0390	0391
0610	0392	0393	0394	0395	0396	0397	0398	0399
0620	0400	0401	0402	0403	0404	0405	0406	0407
0630	0408	0409	0410	0411	0412	0413	0414	0415
0640	0416	0417	0418	0419	0420	0421	0422	0423
0650	0424	0425	0426	0427	0428	0429	0430	0431
0660	0432	0433	0434	0435	0436	0437	0438	0439
0670	0440	0441	0442	0443	0444	0445	0446	0447
0700	0448	0449	0450	0451	0452	0453	0454	0455
0710	0456	0457	0458	0459	0460	0461	0462	0463
0720	0464	0465	0466	0467	0468	0469	0470	0471
0730	0472	0473	0474	0475	0476	0477	0478	0479
0740	0480	0481	0482	0483	0484	0485	0486	0487
0750	0488	0489	0490	0491	0492	0493	0494	0495
0760	0496	0497	0498	0499	0500	0501	0502	0503
0770	0504	0505	0506	0507	0508	0509	0510	0511

1000 to 1777 (Octal)	0512 to 1023 (Decimal)

	0	1	2	3	4	5	6	7
1000	0512	0513	0514	0515	0516	0517	0518	0519
1010	0520	0521	0522	0523	0524	0525	0526	0527
1020	0528	0529	0530	0531	0532	0533	0534	0535
1030	0536	0537	0538	0539	0540	0541	0542	0543
1040	0544	0545	0546	0547	0548	0549	0550	0551
1050	0552	0553	0554	0555	0556	0557	0558	0559
1060	0560	0561	0562	0563	0564	0565	0566	0567
1070	0568	0569	0570	0571	0572	0573	0574	0575
1100	0576	0577	0578	0579	0580	0581	0582	0583
1110	0584	0585	0586	0587	0588	0589	0590	0591
1120	0592	0593	0594	0595	0596	0597	0598	0599
1130	0600	0601	0602	0603	0604	0605	0606	0607
1140	0608	0609	0610	0611	0612	0613	0614	0615
1150	0616	0617	0618	0619	0620	0621	0622	0623
1160	0624	0625	0626	0627	0628	0629	0630	0631
1170	0632	0633	0634	0635	0636	0637	0638	0639
1200	0640	0641	0642	0643	0644	0645	0646	0647
1210	0648	0649	0650	0651	0652	0653	0654	0655
1220	0656	0657	0658	0659	0660	0661	0662	0663
1230	0664	0665	0666	0667	0668	0669	0670	0671
1240	0672	0673	0674	0675	0676	0677	0678	0679
1250	0680	0681	0682	0683	0684	0685	0686	0687
1260	0688	0689	0690	0691	0692	0693	0694	0695
1270	0696	0697	0698	0699	0700	0701	0702	0703
1300	0704	0705	0706	0707	0708	0709	0710	0711
1310	0712	0713	0714	0715	0716	0717	0718	0719
1320	0720	0721	0722	0723	0724	0725	0726	0727
1330	0728	0729	0730	0731	0732	0733	0734	0735
1340	0736	0737	0738	0739	0740	0741	0742	0743
1350	0744	0745	0746	0747	0748	0749	0750	0751
1360	0752	0753	0754	0755	0756	0757	0758	0759
1370	0760	0761	0762	0763	0764	0765	0766	0767

	0	1	2	3	4	5	6	7
1400	0768	0769	0770	0771	0772	0773	0774	0775
1410	0776	0777	0778	0779	0780	0781	0782	0783
1420	0784	0785	0786	0787	0788	0789	0790	0791
1430	0792	0793	0794	0795	0796	0797	0798	0799
1440	0800	0801	0802	0803	0804	0805	0806	0807
1450	0808	0809	0810	0811	0812	0813	0814	0815
1460	0816	0817	0818	0819	0820	0821	0822	0823
1470	0824	0825	0826	0827	0828	0829	0830	0831
1500	0832	0833	0834	0835	0836	0837	0838	0839
1510	0840	0841	0842	0843	0844	0845	0846	0847
1520	0848	0849	0850	0851	0852	0853	0854	0855
1530	0856	0857	0858	0859	0860	0861	0862	0863
1540	0864	0865	0866	0867	0868	0869	0870	0871
1550	0872	0873	0874	0875	0876	0877	0878	0879
1560	0880	0881	0882	0883	0884	0885	0886	0887
1570	0888	0889	0890	0891	0892	0893	0894	0895
1600	0896	0897	0898	0899	0900	0901	0902	0903
1610	0904	0905	0906	0907	0908	0909	0910	0911
1620	0912	0913	0914	0915	0916	0917	0918	0919
1630	0920	0921	0922	0923	0924	0925	0926	0927
1640	0928	0929	0930	0931	0932	0933	0934	0935
1650	0936	0937	0938	0939	0940	0941	0942	0943
1660	0944	0945	0946	0947	0948	0949	0950	0951
1670	0952	0953	0954	0955	0956	0957	0958	0959
1700	0960	0961	0962	0963	0964	0965	0966	0967
1710	0968	0969	0970	0971	0972	0973	0974	0975
1720	0976	0977	0978	0979	0980	0981	0982	0983
1730	0984	0985	0986	0987	0988	0989	0990	0991
1740	0992	0993	0994	0995	0996	0997	0998	0999
1750	1000	1001	1002	1003	1004	1005	1006	1007
1760	1008	1009	1010	1011	1012	1013	1014	1015
1770	1016	1017	1018	1019	1020	1021	1022	1023

Octal-Decimal Integer Conversion Table

	0	1	2	3	4	5	6	7
2000	1024	1025	1026	1027	1028	1029	1030	1031
2010	1032	1033	1034	1035	1036	1037	1038	1039
2020	1040	1041	1042	1043	1044	1045	1046	1047
2030	1048	1049	1050	1051	1052	1053	1054	1055
2040	1056	1057	1058	1059	1060	1061	1062	1063
2050	1064	1065	1066	1067	1068	1069	1070	1071
2060	1072	1073	1074	1075	1076	1077	1078	1079
2070	1080	1081	1082	1083	1084	1085	1086	1087
2100	1088	1089	1090	1091	1092	1093	1094	1095
2110	1096	1097	1098	1099	1100	1101	1102	1103
2120	1104	1105	1106	1107	1108	1109	1110	1111
2130	1112	1113	1114	1115	1116	1117	1118	1119
2140	1120	1121	1122	1123	1124	1125	1126	1127
2150	1128	1129	1130	1131	1132	1133	1134	1135
2160	1136	1137	1138	1139	1140	1141	1142	1143
2170	1144	1145	1146	1147	1148	1149	1150	1151
2200	1152	1153	1154	1155	1156	1157	1158	1159
2210	1160	1161	1162	1163	1164	1165	1166	1167
2220	1168	1169	1170	1171	1172	1173	1174	1175
2230	1176	1177	1178	1179	1180	1181	1182	1183
2240	1184	1185	1186	1187	1188	1189	1190	1191
2250	1192	1193	1194	1195	1196	1197	1198	1199
2260	1200	1201	1202	1203	1204	1205	1206	1207
2270	1208	1209	1210	1211	1212	1213	1214	1215
2300	1216	1217	1218	1219	1220	1221	1222	1223
2310	1224	1225	1226	1227	1228	1229	1230	1231
2320	1232	1233	1234	1235	1236	1237	1238	1239
2330	1240	1241	1242	1243	1244	1245	1246	1247
2340	1248	1249	1250	1251	1252	1253	1254	1255
2350	1256	1257	1258	1259	1260	1261	1262	1263
2360	1264	1265	1266	1267	1268	1269	1270	1271
2370	1272	1273	1274	1275	1276	1277	1278	1279

	0	1	2	3	4	5	6	7
2400	1280	1281	1282	1283	1284	1285	1286	1287
2410	1288	1289	1290	1291	1292	1293	1294	1295
2420	1296	1297	1298	1299	1300	1301	1302	1303
2430	1304	1305	1306	1307	1308	1309	1310	1311
2440	1312	1313	1314	1315	1316	1317	1318	1319
2450	1320	1321	1322	1323	1324	1325	1326	1327
2460	1328	1329	1330	1331	1332	1333	1334	1335
2470	1336	1337	1338	1339	1340	1341	1342	1343
2500	1344	1345	1346	1347	1348	1349	1350	1351
2510	1352	1353	1354	1355	1356	1357	1358	1359
2520	1360	1361	1362	1363	1364	1365	1366	1367
2530	1368	1369	1370	1371	1372	1373	1374	1375
2540	1376	1377	1378	1379	1380	1381	1382	1383
2550	1384	1385	1386	1387	1388	1389	1390	1391
2560	1392	1393	1394	1395	1396	1397	1398	1399
2570	1400	1401	1402	1403	1404	1405	1406	1407
2600	1408	1409	1410	1411	1412	1413	1414	1415
2610	1416	1417	1418	1419	1420	1421	1422	1423
2620	1424	1425	1426	1427	1428	1429	1430	1431
2630	1432	1433	1434	1435	1436	1437	1438	1439
2640	1440	1441	1442	1443	1444	1445	1446	1447
2650	1448	1449	1450	1451	1452	1453	1454	1455
2660	1456	1457	1458	1459	1460	1461	1462	1463
2670	1464	1465	1466	1467	1468	1469	1470	1471
2700	1472	1473	1474	1475	1476	1477	1478	1479
2710	1480	1481	1482	1483	1484	1485	1486	1487
2720	1488	1489	1490	1491	1492	1493	1494	1495
2730	1496	1497	1498	1499	1500	1501	1502	1503
2740	1504	1505	1506	1507	1508	1509	1510	1511
2750	1512	1513	1514	1515	1516	1517	1518	1519
2760	1520	1521	1522	1523	1524	1525	1526	1527
2770	1528	1529	1530	1531	1532	1533	1534	1535

2000	1024
to	to
2777	1535
(Octal)	(Decimal)

Octal	Decimal
10000 -	4096
20000 -	8192
30000 -	12288
40000 -	16384
50000 -	20480
60000 -	24576
70000 -	28672

	0	1	2	3	4	5	6	7
3000	1536	1537	1538	1539	1540	1541	1542	1543
3010	1544	1545	1546	1547	1548	1549	1550	1551
3020	1552	1553	1554	1555	1556	1557	1558	1559
3030	1560	1561	1562	1563	1564	1565	1566	1567
3040	1568	1569	1570	1571	1572	1573	1574	1575
3050	1576	1577	1578	1579	1580	1581	1582	1583
3060	1584	1585	1586	1587	1588	1589	1590	1591
3070	1592	1593	1594	1595	1596	1597	1598	1599
3100	1600	1601	1602	1603	1604	1605	1606	1607
3110	1608	1609	1610	1611	1612	1613	1614	1615
3120	1616	1617	1618	1619	1620	1621	1622	1623
3130	1624	1625	1626	1627	1628	1629	1630	1631
3140	1632	1633	1634	1635	1636	1637	1638	1639
3150	1640	1641	1642	1643	1644	1645	1646	1647
3160	1648	1649	1650	1651	1652	1653	1654	1655
3170	1656	1657	1658	1659	1660	1661	1662	1663
3200	1664	1665	1666	1667	1668	1669	1670	1671
3210	1672	1673	1674	1675	1676	1677	1678	1679
3220	1680	1681	1682	1683	1684	1685	1686	1687
3230	1688	1689	1690	1691	1692	1693	1694	1695
3240	1696	1697	1698	1699	1700	1701	1702	1703
3250	1704	1705	1706	1707	1708	1709	1710	1711
3260	1712	1713	1714	1715	1716	1717	1718	1719
3270	1720	1721	1722	1723	1724	1725	1726	1727
3300	1728	1729	1730	1731	1732	1733	1734	1735
3310	1736	1737	1738	1739	1740	1741	1742	1743
3320	1744	1745	1746	1747	1748	1749	1750	1751
3330	1752	1753	1754	1755	1756	1757	1758	1759
3340	1760	1761	1762	1763	1764	1765	1766	1767
3350	1768	1769	1770	1771	1772	1773	1774	1775
3360	1776	1777	1778	1779	1780	1781	1782	1783
3370	1784	1785	1786	1787	1788	1789	1790	1791

	0	1	2	3	4	5	6	7
3400	1792	1793	1794	1795	1796	1797	1798	1799
3410	1800	1801	1802	1803	1804	1805	1806	1807
3420	1808	1809	1810	1811	1812	1813	1814	1815
3430	1816	1817	1818	1819	1820	1821	1822	1823
3440	1824	1825	1826	1827	1828	1829	1830	1831
3450	1832	1833	1834	1835	1836	1837	1838	1839
3460	1840	1841	1842	1843	1844	1845	1846	1847
3470	1848	1849	1850	1851	1852	1853	1854	1855
3500	1856	1857	1858	1859	1860	1861	1862	1863
3510	1864	1865	1866	1867	1868	1869	1870	1871
3520	1872	1873	1874	1875	1876	1877	1878	1879
3530	1880	1881	1882	1883	1884	1885	1886	1887
3540	1888	1889	1890	1891	1892	1893	1894	1895
3550	1896	1897	1898	1899	1900	1901	1902	1903
3560	1904	1905	1906	1907	1908	1909	1910	1911
3570	1912	1913	1914	1915	1916	1917	1918	1919
3600	1920	1921	1922	1923	1924	1925	1926	1927
3610	1928	1929	1930	1931	1932	1933	1934	1935
3620	1936	1937	1938	1939	1940	1941	1942	1943
3630	1944	1945	1946	1947	1948	1949	1950	1951
3640	1952	1953	1954	1955	1956	1957	1958	1959
3650	1960	1961	1962	1963	1964	1965	1966	1967
3660	1968	1969	1970	1971	1972	1973	1974	1975
3670	1976	1977	1978	1979	1980	1981	1982	1983
3700	1984	1985	1986	1987	1988	1989	1990	1991
3710	1992	1993	1994	1995	1996	1997	1998	1999
3720	2000	2001	2002	2003	2004	2005	2006	2007
3730	2008	2009	2010	2011	2012	2013	2014	2015
3740	2016	2017	2018	2019	2020	2021	2022	2023
3750	2024	2025	2026	2027	2028	2029	2030	2031
3760	2032	2033	2034	2035	2036	2037	2038	2039
3770	2040	2041	2042	2043	2044	2045	2046	2047

3000	1536
to	to
3777	2047
(Octal)	(Decimal)

Octal-Decimal Integer Conversion Table

4000 to 4777 (Octal)	2048 to 2559 (Decimal)

Octal	Decimal
10000 -	4096
20000 -	8192
30000 -	12288
40000 -	16384
50000 -	20480
60000 -	24576
70000 -	28672

	0	1	2	3	4	5	6	7
4000	2048	2049	2050	2051	2052	2053	2054	2055
4010	2056	2057	2058	2059	2060	2061	2062	2063
4020	2064	2065	2066	2067	2068	2069	2070	2071
4030	2072	2073	2074	2075	2076	2077	2078	2079
4040	2080	2081	2082	2083	2084	2085	2086	2087
4050	2088	2089	2090	2091	2092	2093	2094	2095
4060	2096	2097	2098	2099	2100	2101	2102	2103
4070	2104	2105	2106	2107	2108	2109	2110	2111
4100	2112	2113	2114	2115	2116	2117	2118	2119
4110	2120	2121	2122	2123	2124	2125	2126	2127
4120	2128	2129	2130	2131	2132	2133	2134	2135
4130	2136	2137	2138	2139	2140	2141	2142	2143
4140	2144	2145	2146	2147	2148	2149	2150	2151
4150	2152	2153	2154	2155	2156	2157	2158	2159
4160	2160	2161	2162	2163	2164	2165	2166	2167
4170	2168	2169	2170	2171	2172	2173	2174	2175
4200	2176	2177	2178	2179	2180	2181	2182	2183
4210	2184	2185	2186	2187	2188	2189	2190	2191
4220	2192	2193	2194	2195	2196	2197	2198	2199
4230	2200	2201	2202	2203	2204	2205	2206	2207
4240	2208	2209	2210	2211	2212	2213	2214	2215
4250	2216	2217	2218	2219	2220	2221	2222	2223
4260	2224	2225	2226	2227	2228	2229	2230	2231
4270	2232	2233	2234	2235	2236	2237	2238	2239
4300	2240	2241	2242	2243	2244	2245	2246	2247
4310	2248	2249	2250	2251	2252	2253	2254	2255
4320	2256	2257	2258	2259	2260	2261	2262	2263
4330	2264	2265	2266	2267	2268	2269	2270	2271
4340	2272	2273	2274	2275	2276	2277	2278	2279
4350	2280	2281	2282	2283	2284	2285	2286	2287
4360	2288	2289	2290	2291	2292	2293	2294	2295
4370	2296	2297	2298	2299	2300	2301	2302	2303

	0	1	2	3	4	5	6	7
4400	2304	2305	2306	2307	2308	2309	2310	2311
4410	2312	2313	2314	2315	2316	2317	2318	2319
4420	2320	2321	2322	2323	2324	2325	2326	2327
4430	2328	2329	2330	2331	2332	2333	2334	2335
4440	2336	2337	2338	2339	2340	2341	2342	2343
4450	2344	2345	2346	2347	2348	2349	2350	2351
4460	2352	2353	2354	2355	2356	2357	2358	2359
4470	2360	2361	2362	2363	2364	2365	2366	2367
4500	2368	2369	2370	2371	2372	2373	2374	2375
4510	2376	2377	2378	2379	2380	2381	2382	2383
4520	2384	2385	2386	2387	2388	2389	2390	2391
4530	2392	2393	2394	2395	2396	2397	2398	2399
4540	2400	2401	2402	2403	2404	2405	2406	2407
4550	2408	2409	2410	2411	2412	2413	2414	2415
4560	2416	2417	2418	2419	2420	2421	2422	2423
4570	2424	2425	2426	2427	2428	2429	2430	2431
4600	2432	2433	2434	2435	2436	2437	2438	2439
4610	2440	2441	2442	2443	2444	2445	2446	2447
4620	2448	2449	2450	2451	2452	2453	2454	2455
4630	2456	2457	2458	2459	2460	2461	2462	2463
4640	2464	2465	2466	2467	2468	2469	2470	2471
4650	2472	2473	2474	2475	2476	2477	2478	2479
4660	2480	2481	2482	2483	2484	2485	2486	2487
4670	2488	2489	2490	2491	2492	2493	2494	2495
4700	2496	2497	2498	2499	2500	2501	2502	2503
4710	2504	2505	2506	2507	2508	2509	2510	2511
4720	2512	2513	2514	2515	2516	2517	2518	2519
4730	2520	2521	2522	2523	2524	2525	2526	2527
4740	2528	2529	2530	2531	2532	2533	2534	2535
4750	2536	2537	2538	2539	2540	2541	2542	2543
4760	2544	2545	2546	2547	2548	2549	2550	2551
4770	2552	2553	2554	2555	2556	2557	2558	2559

5000 to 5777 (Octal)	2560 to 3071 (Decimal)

	0	1	2	3	4	5	6	7
5000	2560	2561	2562	2563	2564	2565	2566	2567
5010	2568	2569	2570	2571	2572	2573	2574	2575
5020	2576	2577	2578	2579	2580	2581	2582	2583
5030	2584	2585	2586	2587	2588	2589	2590	2591
5040	2592	2593	2594	2595	2596	2597	2598	2599
5050	2600	2601	2602	2603	2604	2605	2606	2607
5060	2608	2609	2610	2611	2612	2613	2614	2615
5070	2616	2617	2618	2619	2620	2621	2622	2623
5100	2624	2625	2626	2627	2628	2629	2630	2631
5110	2632	2633	2634	2635	2636	2637	2638	2639
5120	2640	2641	2642	2643	2644	2645	2646	2647
5130	2648	2649	2650	2651	2652	2653	2654	2655
5140	2656	2657	2658	2659	2660	2661	2662	2663
5150	2664	2665	2666	2667	2668	2669	2670	2671
5160	2672	2673	2674	2675	2676	2677	2678	2679
5170	2680	2681	2682	2683	2684	2685	2686	2687
5200	2688	2689	2690	2691	2692	2693	2694	2695
5210	2696	2697	2698	2699	2700	2701	2702	2703
5220	2704	2705	2706	2707	2708	2709	2710	2711
5230	2712	2713	2714	2715	2716	2717	2718	2719
5240	2720	2721	2722	2723	2724	2725	2726	2727
5250	2728	2729	2730	2731	2732	2733	2734	2735
5260	2736	2737	2738	2739	2740	2741	2742	2743
5270	2744	2745	2746	2747	2748	2749	2750	2751
5300	2752	2753	2754	2755	2756	2757	2758	2759
5310	2760	2761	2762	2763	2764	2765	2766	2767
5320	2768	2769	2770	2771	2772	2773	2774	2775
5330	2776	2777	2778	2779	2780	2781	2782	2783
5340	2784	2785	2786	2787	2788	2789	2790	2791
5350	2792	2793	2794	2795	2796	2797	2798	2799
5360	2800	2801	2802	2803	2804	2805	2806	2807
5370	2808	2809	2810	2811	2812	2813	2814	2815

	0	1	2	3	4	5	6	7
5400	2816	2817	2818	2819	2820	2821	2822	2823
5410	2824	2825	2826	2827	2828	2829	2830	2831
5420	2832	2833	2834	2835	2836	2837	2838	2839
5430	2840	2841	2842	2843	2844	2845	2846	2847
5440	2848	2849	2850	2851	2852	2853	2854	2855
5450	2856	2857	2858	2859	2860	2861	2862	2863
5460	2864	2865	2866	2867	2868	2869	2870	2871
5470	2872	2873	2874	2875	2876	2877	2878	2879
5500	2880	2881	2882	2883	2884	2885	2886	2887
5510	2888	2889	2890	2891	2892	2893	2894	2895
5520	2896	2897	2898	2899	2900	2901	2902	2903
5530	2904	2905	2906	2907	2908	2909	2910	2911
5540	2912	2913	2914	2915	2916	2917	2918	2919
5550	2920	2921	2922	2923	2924	2925	2926	2927
5560	2928	2929	2930	2931	2932	2933	2934	2935
5570	2936	2937	2938	2939	2940	2941	2942	2943
5600	2944	2945	2946	2947	2948	2949	2950	2951
5610	2952	2953	2954	2955	2956	2957	2958	2959
5620	2960	2961	2962	2963	2964	2965	2966	2967
5630	2968	2969	2970	2971	2972	2973	2974	2975
5640	2976	2977	2978	2979	2980	2981	2982	2983
5650	2984	2985	2986	2987	2988	2989	2990	2991
5660	2992	2993	2994	2995	2996	2997	2998	2999
5670	3000	3001	3002	3003	3004	3005	3006	3007
5700	3008	3009	3010	3011	3012	3013	3014	3015
5710	3016	3017	3018	3019	3020	3021	3022	3023
5720	3024	3025	3026	3027	3028	3029	3030	3031
5730	3032	3033	3034	3035	3036	3037	3038	3039
5740	3040	3041	3042	3043	3044	3045	3046	3047
5750	3048	3049	3050	3051	3052	3053	3054	3055
5760	3056	3057	3058	3059	3060	3061	3062	3063
5770	3064	3065	3066	3067	3068	3069	3070	3071

Octal-Decimal Integer Conversion Table

	0	1	2	3	4	5	6	7
6000	3072	3073	3074	3075	3076	3077	3078	3079
6010	3080	3081	3082	3083	3084	3085	3086	3087
6020	3088	3089	3090	3091	3092	3093	3094	3095
6030	3096	3097	3098	3099	3100	3101	3102	3103
6040	3104	3105	3106	3107	3108	3109	3110	3111
6050	3112	3113	3114	3115	3116	3117	3118	3119
6060	3120	3121	3122	3123	3124	3125	3126	3127
6070	3128	3129	3130	3131	3132	3133	3134	3135
6100	3136	3137	3138	3139	3140	3141	3142	3143
6110	3144	3145	3146	3147	3148	3149	3150	3151
6120	3152	3153	3154	3155	3156	3157	3158	3159
6130	3160	3161	3162	3163	3164	3165	3166	3167
6140	3168	3169	3170	3171	3172	3173	3174	3175
6150	3176	3177	3178	3179	3180	3181	3182	3183
6160	3184	3185	3186	3187	3188	3189	3190	3191
6170	3192	3193	3194	3195	3196	3197	3198	3199
6200	3200	3201	3202	3203	3204	3205	3206	3207
6210	3208	3209	3210	3211	3212	3213	3214	3215
6220	3216	3217	3218	3219	3220	3221	3222	3223
6230	3224	3225	3226	3227	3228	3229	3230	3231
6240	3232	3233	3234	3235	3236	3237	3238	3239
6250	3240	3241	3242	3243	3244	3245	3246	3247
6260	3248	3249	3250	3251	3252	3253	3254	3255
6270	3256	3257	3258	3259	3260	3261	3262	3263
6300	3264	3265	3266	3267	3268	3269	3270	3271
6310	3272	3273	3274	3275	3276	3277	3278	3279
6320	3280	3281	3282	3283	3284	3285	3286	3287
6330	3288	3289	3290	3291	3292	3293	3294	3295
6340	3296	3297	3298	3299	3300	3301	3302	3303
6350	3304	3305	3306	3307	3308	3309	3310	3311
6360	3312	3313	3314	3315	3316	3317	3318	3319
6370	3320	3321	3322	3323	3324	3325	3326	3327

	0	1	2	3	4	5	6	7
6400	3328	3329	3330	3331	3332	3333	3334	3335
6410	3336	3337	3338	3339	3340	3341	3342	3343
6420	3344	3345	3346	3347	3348	3349	3350	3351
6430	3352	3353	3354	3355	3356	3357	3358	3359
6440	3360	3361	3362	3363	3364	3365	3366	3367
6450	3368	3369	3370	3371	3372	3373	3374	3375
6460	3376	3377	3378	3379	3380	3381	3382	3383
6470	3384	3385	3386	3387	3388	3389	3390	3391
6500	3392	3393	3394	3395	3396	3397	3398	3399
6510	3400	3401	3402	3403	3404	3405	3406	3407
6520	3408	3409	3410	3411	3412	3413	3414	3415
6530	3416	3417	3418	3419	3420	3421	3422	3423
6540	3424	3425	3426	3427	3428	3429	3430	3431
6550	3432	3433	3434	3435	3436	3437	3438	3439
6560	3440	3441	3442	3443	3444	3445	3446	3447
6570	3448	3449	3450	3451	3452	3453	3454	3455
6600	3456	3457	3458	3459	3460	3461	3462	3463
6610	3464	3465	3466	3467	3468	3469	3470	3471
6620	3472	3473	3474	3475	3476	3477	3478	3479
6630	3480	3481	3482	3483	3484	3485	3486	3487
6640	3488	3489	3490	3491	3492	3493	3494	3495
6650	3496	3497	3498	3499	3500	3501	3502	3503
6660	3504	3505	3506	3507	3508	3509	3510	3511
6670	3512	3513	3514	3515	3516	3517	3518	3519
6700	3520	3521	3522	3523	3524	3525	3526	3527
6710	3528	3529	3530	3531	3532	3533	3534	3535
6720	3536	3537	3538	3539	3540	3541	3542	3543
6730	3544	3545	3546	3547	3548	3549	3550	3551
6740	3552	3553	3554	3555	3556	3557	3558	3559
6750	3560	3561	3562	3563	3564	3565	3566	3567
6760	3568	3569	3570	3571	3572	3573	3574	3575
6770	3576	3577	3578	3579	3580	3581	3582	3583

6000 to 6777 (Octal)	3072 to 3583 (Decimal)

Octal	Decimal
10000 -	4096
20000 -	8192
30000 -	12288
40000 -	16384
50000 -	20480
60000 -	24576
70000 -	28672

	0	1	2	3	4	5	6	7
7000	3584	3585	3586	3587	3588	3589	3590	3591
7010	3592	3593	3594	3595	3596	3597	3598	3599
7020	3600	3601	3602	3603	3604	3605	3606	3607
7030	3608	3609	3610	3611	3612	3613	3614	3615
7040	3616	3617	3618	3619	3620	3621	3622	3623
7050	3624	3625	3626	3627	3628	3629	3630	3631
7060	3632	3633	3634	3635	3636	3637	3638	3639
7070	3640	3641	3642	3643	3644	3645	3646	3647
7100	3648	3649	3650	3651	3652	3653	3654	3655
7110	3656	3657	3658	3659	3660	3661	3662	3663
7120	3664	3665	3666	3667	3668	3669	3670	3671
7130	3672	3673	3674	3675	3676	3677	3678	3679
7140	3680	3681	3682	3683	3684	3685	3686	3687
7150	3688	3689	3690	3691	3692	3693	3694	3695
7160	3696	3697	3698	3699	3700	3701	3702	3703
7170	3704	3705	3706	3707	3708	3709	3710	3711
7200	3712	3713	3714	3715	3716	3717	3718	3719
7210	3720	3721	3722	3723	3724	3725	3726	3727
7220	3728	3729	3730	3731	3732	3733	3734	3735
7230	3736	3737	3738	3739	3740	3741	3742	3743
7240	3744	3745	3746	3747	3748	3749	3750	3751
7250	3752	3753	3754	3755	3756	3757	3758	3759
7260	3760	3761	3762	3763	3764	3765	3766	3767
7270	3768	3769	3770	3771	3772	3773	3774	3775
7300	3776	3777	3778	3779	3780	3781	3782	3783
7310	3784	3785	3786	3787	3788	3789	3790	3791
7320	3792	3793	3794	3795	3796	3797	3798	3799
7330	3800	3801	3802	3803	3804	3805	3806	3807
7340	3808	3809	3810	3811	3812	3813	3814	3815
7350	3816	3817	3818	3819	3820	3821	3822	3823
7360	3824	3825	3826	3827	3828	3829	3830	3831
7370	3832	3833	3834	3835	3836	3837	3838	3839

	0	1	2	3	4	5	6	7
7400	3840	3841	3842	3843	3844	3845	3846	3847
7410	3848	3849	3850	3851	3852	3853	3854	3855
7420	3856	3857	3858	3859	3860	3861	3862	3863
7430	3864	3865	3866	3867	3868	3869	3870	3871
7440	3872	3873	3874	3875	3876	3877	3878	3879
7450	3880	3881	3882	3883	3884	3885	3886	3887
7460	3888	3889	3890	3891	3892	3893	3894	3895
7470	3896	3897	3898	3899	3900	3901	3902	3903
7500	3904	3905	3906	3907	3908	3909	3910	3911
7510	3912	3913	3914	3915	3916	3917	3918	3919
7520	3920	3921	3922	3923	3924	3925	3926	3927
7530	3928	3929	3930	3931	3932	3933	3934	3935
7540	3936	3937	3938	3939	3940	3941	3942	3943
7550	3944	3945	3946	3947	3948	3949	3950	3951
7560	3952	3953	3954	3955	3956	3957	3958	3959
7570	3960	3961	3962	3963	3964	3965	3966	3967
7600	3968	3969	3970	3971	3972	3973	3974	3975
7610	3976	3977	3978	3979	3980	3981	3982	3983
7620	3984	3985	3986	3987	3988	3989	3990	3991
7630	3992	3993	3994	3995	3996	3997	3998	3999
7640	4000	4001	4002	4003	4004	4005	4006	4007
7650	4008	4009	4010	4011	4012	4013	4014	4015
7660	4016	4017	4018	4019	4020	4021	1022	4023
7670	4024	4025	4026	4027	4028	4029	4030	4031
7700	4032	4033	4034	4035	4036	4037	4038	4039
7710	4040	4041	4042	4043	4044	4045	4046	4047
7720	4048	4049	4050	4051	4052	4053	4054	4055
7730	4056	4057	4058	4059	4060	4061	4062	4063
7740	4064	4065	4066	4067	4068	4069	4070	4071
7750	4072	4073	4074	4075	4076	4077	4078	4079
7760	4080	4081	4082	4083	4084	4085	4086	4087
7770	4088	4089	4090	4091	4092	4093	4094	4095

7000 to 7777 (Octal)	3584 to 4095 (Decimal)

OCTAL - DECIMAL FRACTION CONVERSION TABLE

OCTAL	DEC.	OCTAL	DEC.	OCTAL	DEC.	OCTAL	DEC.
.000	.000000	.100	.125000	.200	.250000	.300	.375000
.001	.001953	.101	.126953	.201	.251953	.301	.376953
.002	.003906	.102	.128906	.202	.253906	.302	.378906
.003	.005859	.103	.130859	.203	.255859	.303	.380859
.004	.007812	.104	.132812	.204	.257812	.304	.382812
.005	.009765	.105	.134765	.205	.259765	.305	.384765
.006	.011718	.106	.136718	.206	.261718	.306	.386718
.007	.013671	.107	.138671	.207	.263671	.307	.388671
.010	.015625	.110	.140625	.210	.265625	.310	.390625
.011	.017578	.111	.142578	.211	.267578	.311	.392578
.012	.019531	.112	.144531	.212	.269531	.312	.394531
.013	.021484	.113	.146484	.213	.271484	.313	.396484
.014	.023437	.114	.148437	.214	.273437	.314	.398437
.015	.025390	.115	.150390	.215	.275390	.315	.400390
.016	.027343	.116	.152343	.216	.277343	.316	.402343
.017	.029296	.117	.154296	.217	.279296	.317	.404296
.020	.031250	.120	.156250	.220	.281250	.320	.406250
.021	.033203	.121	.158203	.221	.283203	.321	.408203
.022	.035156	.122	.160156	.222	.285156	.322	.410156
.023	.037109	.123	.162109	.223	.287109	.323	.412109
.024	.039062	.124	.164062	.224	.289062	.324	.414062
.025	.041015	.125	.166015	.225	.291015	.325	.416015
.026	.042968	.126	.167968	.226	.292968	.326	.417968
.027	.044921	.127	.169921	.227	.294921	.327	.419921
.030	.046875	.130	.171875	.230	.296875	.330	.421875
.031	.048828	.131	.173828	.231	.298828	.331	.423828
.032	.050781	.132	.175781	.232	.300781	.332	.425781
.033	.052734	.133	.177734	.233	.302734	.333	.427734
.034	.054687	.134	.179687	.234	.304687	.334	.429687
.035	.056640	.135	.181640	.235	.306640	.335	.431640
.036	.058593	.136	.183593	.236	.308593	.336	.433593
.037	.060546	.137	.185546	.237	.310546	.337	.435546
.040	.062500	.140	.187500	.240	.312500	.340	.437500
.041	.064453	.141	.189453	.241	.314453	.341	.439453
.042	.066406	.142	.191406	.242	.316406	.342	.441406
.043	.068359	.143	.193359	.243	.318359	.343	.443359
.044	.070312	.144	.195312	.244	.320312	.344	.445312
.045	.072265	.145	.197265	.245	.322265	.345	.447265
.046	.074218	.146	.199218	.246	.324218	.346	.449218
.047	.076171	.147	.201171	.247	.326171	.347	.451171
.050	.078125	.150	.203125	.250	.328125	.350	.453125
.051	.080078	.151	.205078	.251	.330078	.351	.455078
.052	.082031	.152	.207031	.252	.332031	.352	.457031
.053	.083984	.153	.208984	.253	.333984	.353	.458984
.054	.085937	.154	.210937	.254	.335937	.354	.460937
.055	.087890	.155	.212890	.255	.337890	.355	.462890
.056	.089843	.156	.214843	.256	.339843	.356	.464843
.057	.091796	.157	.216796	.257	.341796	.357	.466796
.060	.093750	.160	.218750	.260	.343750	.360	.468750
.061	.095703	.161	.220703	.261	.345703	.361	.470703
.062	.097656	.162	.222656	.262	.347656	.362	.472656
.063	.099609	.163	.224609	.263	.349609	.363	.474609
.064	.101562	.164	.226562	.264	.351562	.364	.476562
.065	.103515	.165	.228515	.265	.353515	.365	.478515
.066	.105468	.166	.230468	.266	.355468	.366	.480468
.067	.107421	.167	.232421	.267	.357421	.367	.482421
.070	.109375	.170	.234375	.270	.359375	.370	.484375
.071	.111328	.171	.236328	.271	.361328	.371	.486328
.072	.113281	.172	.238281	.272	.363281	.372	.488281
.073	.115234	.173	.240234	.273	.365234	.373	.490234
.074	.117187	.174	.242187	.274	.367187	.374	.492187
.075	.119140	.175	.244140	.275	.369140	.375	.494140
.076	.121093	.176	.246093	.276	.371093	.376	.496093
.077	.123046	.177	.248046	.277	.373046	.377	.498046

Octal-Decimal Fraction Conversion Table

OCTAL	DEC.	OCTAL	DEC.	OCTAL	DEC.	OCTAL	DEC.
.000000	.000000	.000100	.000244	.000200	.000488	.000300	.000732
.000001	.000003	.000101	.000247	.000201	.000492	.000301	.000736
.000002	.000007	.000102	.000251	.000202	.000495	.000302	.000740
.000003	.000011	.000103	.000255	.000203	.000499	.000303	.000743
.000004	.000015	.000104	.000259	.000204	.000503	.000304	.000747
.000005	.000019	.000105	.000263	.000205	.000507	.000305	.000751
.000006	.000022	.000106	.000267	.000206	.000511	.000306	.000755
.000007	.000026	.000107	.000270	.000207	.000514	.000307	.000759
.000010	.000030	.000110	.000274	.000210	.000518	.000310	.000762
.000011	.000034	.000111	.000278	.000211	.000522	.000311	.000766
.000012	.000038	.000112	.000282	.000212	.000526	.000312	.000770
.000013	.000041	.000113	.000286	.000213	.000530	.000313	.000774
.000014	.000045	.000114	.000289	.000214	.000534	.000314	.000778
.000015	.000049	.000115	.000293	.000215	.000537	.000315	.000782
.000016	.000053	.000116	.000297	.000216	.000541	.000316	.000785
.000017	.000057	.000117	.000301	.000217	.000545	.000317	.000789
.000020	.000061	.000120	.000305	.000220	.000549	.000320	.000793
.000021	.000064	.000121	.000308	.000221	.000553	.000321	.000797
.000022	.000068	.000122	.000312	.000222	.000556	.000322	.000801
.000023	.000072	.000123	.000316	.000223	.000560	.000323	.000805
.000024	.000076	.000124	.000320	.000224	.000564	.000324	.000808
.000025	.000080	.000125	.000324	.000225	.000568	.000325	.000812
.000026	.000083	.000126	.000328	.000226	.000572	.000326	.000816
.000027	.000087	.000127	.000331	.000227	.000576	.000327	.000820
.000030	.000091	.000130	.000335	.000230	.000579	.000330	.000823
.000031	.000095	.000131	.000339	.000231	.000583	.000331	.000827
.000032	.000099	.000132	.000343	.000232	.000587	.000332	.000831
.000033	.000102	.000133	.000347	.000233	.000591	.000333	.000835
.000034	.000106	.000134	.000350	.000234	.000595	.000334	.000839
.000035	.000110	.000135	.000354	.000235	.000598	.000335	.000843
.000036	.000114	.000136	.000358	.000236	.000602	.000336	.000846
.000037	.000118	.000137	.000362	.000237	.000606	.000337	.000850
.000040	.000122	.000140	.000366	.000240	.000610	.000340	.000854
.000041	.000125	.000141	.000370	.000241	.000614	.000341	.000858
.000042	.000129	.000142	.000373	.000242	.000617	.000342	.000862
.000043	.000133	.000143	.000377	.000243	.000621	.000343	.000865
.000044	.000137	.000144	.000381	.000244	.000625	.000344	.000869
.000045	.000141	.000145	.000385	.000245	.000629	.000345	.000873
.000046	.000144	.000146	.000389	.000246	.000633	.000346	.000877
.000047	.000148	.000147	.000392	.000247	.000637	.000347	.000881
.000050	.000152	.000150	.000396	.000250	.000640	.000350	.000885
.000051	.000156	.000151	.000400	.000251	.000644	.000351	.000888
.000052	.000160	.000152	.000404	.000252	.000648	.000352	.000892
.000053	.000164	.000153	.000408	.000253	.000652	.000353	.000896
.000054	.000167	.000154	.000411	.000254	.000656	.000354	.000900
.000055	.000171	.000155	.000415	.000255	.000659	.000355	.000904
.000056	.000175	.000156	.000419	.000256	.000663	.000356	.000907
.000057	.000179	.000157	.000423	.000257	.000667	.000357	.000911
.000060	.000183	.000160	.000427	.000260	.000671	.000360	.000915
.000061	.000186	.000161	.000431	.000261	.000675	.000361	.000919
.000062	.000190	.000162	.000434	.000262	.000679	.000362	.000923
.000063	.000194	.000163	.000438	.000263	.000682	.000363	.000926
.000064	.000198	.000164	.000442	.000264	.000686	.000364	.000930
.000065	.000202	.000165	.000446	.000265	.000690	.000365	.000934
.000066	.000205	.000166	.000450	.000266	.000694	.000366	.000938
.000067	.000209	.000167	.000453	.000267	.000698	.000367	.000942
.000070	.000213	.000170	.000457	.000270	.000701	.000370	.000946
.000071	.000217	.000171	.000461	.000271	.000705	.000371	.000949
.000072	.000221	.000172	.000465	.000272	.000709	.000372	.000953
.000073	.000225	.000173	.000469	.000273	.000713	.000373	.000957
.000074	.000228	.000174	.000473	.000274	.000717	.000374	.000961
.000075	.000232	.000175	.000476	.000275	.000720	.000375	.000965
.000076	.000236	.000176	.000480	.000276	.000724	.000376	.000968
.000077	.000240	.000177	.000484	.000277	.000728	.000377	.000972

Octal-Decimal Fraction Conversion Table

OCTAL	DEC.	OCTAL	DEC.	OCTAL	DEC.	OCTAL	DEC.
.000400	.000976	.000500	.001220	.000600	.001464	.000700	.001708
.000401	.000980	.000501	.001224	.000601	.001468	.000701	.001712
.000402	.000984	.000502	.001228	.000602	.001472	.000702	.001716
.000403	.000988	.000503	.001232	.000603	.001476	.000703	.001720
.000404	.000991	.000504	.001235	.000604	.001480	.000704	.001724
.000405	.000995	.000505	.001239	.000605	.001483	.000705	.001728
.000406	.000999	.000506	.001243	.000606	.001487	.000706	.001731
.000407	.001003	.000507	.001247	.000607	.001491	.000707	.001735
.000410	.001007	.000510	.001251	.000610	.001495	.000710	.001739
.000411	.001010	.000511	.001255	.000611	.001499	.000711	.001743
.000412	.001014	.000512	.001258	.000612	.001502	.000712	.001747
.000413	.001018	.000513	.001262	.000613	.001506	.000713	.001750
.000414	.001022	.000514	.001266	.000614	.001510	.000714	.001754
.000415	.001026	.000515	.001270	.000615	.001514	.000715	.001758
.000416	.001029	.000516	.001274	.000616	.001518	.000716	.001762
.000417	.001033	.000517	.001277	.000617	.001522	.000717	.001766
.000420	.001037	.000520	.001281	.000620	.001525	.000720	.001770
.000421	.001041	.000521	.001285	.000621	.001529	.000721	.001773
.000422	.001045	.000522	.001289	.000622	.001533	.000722	.001777
.000423	.001049	.000523	.001293	.000623	.001537	.000723	.001781
.000424	.001052	.000524	.001296	.000624	.001541	.000724	.001785
.000425	.001056	.000525	.001300	.000625	.001544	.000725	.001789
.000426	.001060	.000526	.001304	.000626	.001548	.000726	.001792
.000427	.001064	.000527	.001308	.000627	.001552	.000727	.001796
.000430	.001068	.000530	.001312	.000630	.001556	.000730	.001800
.000431	.001071	.000531	.001316	.000631	.001560	.000731	.001804
.000432	.001075	.000532	.001319	.000632	.001564	.000732	.001808
.000433	.001079	.000533	.001323	.000633	.001567	.000733	.001811
.000434	.001083	.000534	.001327	.000634	.001571	.000734	.001815
.000435	.001087	.000535	.001331	.000635	.001575	.000735	.001819
.000436	.001091	.000536	.001335	.000636	.001579	.000736	.001823
.000437	.001094	.000537	.001338	.000637	.001583	.000737	.001827
.000440	.001098	.000540	.001342	.000640	.001586	.000740	.001831
.000441	.001102	.000541	.001346	.000641	.001590	.000741	.001834
.000442	.001106	.000542	.001350	.000642	.001594	.000742	.001838
.000443	.001110	.000543	.001354	.000643	.001598	.000743	.001842
.000444	.001113	.000544	.001358	.000644	.001602	.000744	.001846
.000445	.001117	.000545	.001361	.000645	.001605	.000745	.001850
.000446	.001121	.000546	.001365	.000646	.001609	.000746	.001853
.000447	.001125	.000547	.001369	.000647	.001613	.000747	.001857
.000450	.001129	.000550	.001373	.000650	.001617	.000750	.001861
.000451	.001132	.000551	.001377	.000651	.001621	.000751	.001865
.000452	.001136	.000552	.001380	.000652	.001625	.000752	.001869
.000453	.001140	.000553	.001384	.000653	.001628	.000753	.001873
.000454	.001144	.000554	.001388	.000654	.001632	.000754	.001876
.000455	.001148	.000555	.001392	.000655	.001636	.000755	.001880
.000456	.001152	.000556	.001396	.000656	.001640	.000756	.001884
.000457	.001155	.000557	.001399	.000657	.001644	.000757	.001888
.000460	.001159	.000560	.001403	.000660	.001647	.000760	.001892
.000461	.001163	.000561	.001407	.000661	.001651	.000761	.001895
.000462	.001167	.000562	.001411	.000662	.001655	.000762	.001899
.000463	.001171	.000563	.001415	.000663	.001659	.000763	.001903
.000464	.001174	.000564	.001419	.000664	.001663	.000764	.001907
.000465	.001178	.000565	.001422	.000665	.001667	.000765	.001911
.000466	.001182	.000566	.001426	.000666	.001670	.000766	.001914
.000467	.001186	.000567	.001430	.000667	.001674	.000767	.001918
.000470	.001190	.000570	.001434	.000670	.001678	.000770	.001922
.000471	.001194	.000571	.001438	.000671	.001682	.000771	.001926
.000472	.001197	.000572	.001441	.000672	.001686	.000772	.001930
.000473	.001201	.000573	.001445	.000673	.001689	.000773	.001934
.000474	.001205	.000574	.001449	.000674	.001693	.000774	.001937
.000475	.001209	.000575	.001453	.000675	.001697	.000775	.001941
.000476	.001213	.000576	.001457	.000676	.001701	.000776	.001945
.000477	.001216	.000577	.001461	.000677	.001705	.000777	.001949

TWO'S COMPLEMENT ARITHMETIC

SDS computer systems hold numbers in memory in two's complement form. Single-precision numbers have 23 magnitude bits and a sign bit. The sign bit is in the first bit position to the left of the most significant of the magnitude bits. Thus, the sign bit actually is a part of the number in all arithmetic operations. A "0" bit denotes a positive sign and a "1" bit denotes a negative sign. In this system, the negative of a number is its two's complement.

An algorithm for finding the two's complement of a binary number with attached sign bit is:

> To find the two's complement of the binary number B that has n significant bits including the sign bit, subtract it from the number 2^n expressed in binary form. This latter number is a "1" followed by n zeros.

EXAMPLES:

> The following example indicates the two's complement of binary numbers held in five bits plus a sign bit. Their decimal equivalents are on the left.

Decimal Number	Binary Equivalent	Negative of Decimal Number	Two's Complement of Binary Equivalent
+ 2	000010	- 2	111110
+14	001110	-14	110010

> In the addition example below, decimal notation is on the left and binary notation on the right.

+20	010100
-03	111101
+17	010001

In the computer, 24-bit numbers are written as eight octal digits for convenience. The following example shows three forms of the same addition -- decimal, binary, and octal, respectively. The binary number is assumed to be an integer.

Decimal	Binary	Octal
+21	000000000000000000010101	00000025
-03	111111111111111111111101	77777775
+18	000000000000000000010010	00000022

As the examples indicate, the sign bit is an integral part of the number to which it is attached and its value, plus or minus, is automatically taken care of during the use of two's complement arithmetic. This property is used when numbers of different length are added. For example, assume that these two signed, two's complemented, negative numbers of 6-bit and 3-bit length are added:

Decimal	Binary
-21	101011
-03	101
-24	110000 = -16_{10}

Notice that the third least significant bit of the first number is added to the sign bit of the second number causing an erroneous result. This error is corrected by filling in the empty, most significant bit positions with the value of the sign bit of the shorter number:

Decimal	Binary
-21	101011
-03	111101
-24	101000 = -24_{10}

This property suggests:

1) Filling the empty bit positions with the sign value of a positive number, that is, zeros, has no changing effect on the result, and

2) If the two's complement is taken by the method suggested, where n is the larger number's length, the sign value is automatically appended to the smaller number. For instance, in the above example, if the complement of 03 is taken using n = 6,

```
1000000
    011
111101
```

the sign is properly appended to the number.

This procedure is called "extending" the sign of a number.

OPTIONAL EQUIPMENT

REAL-TIME CLOCK

The Real-Time Clock (RTC) provides a highly flexible time-orientation system for the SDS 92 Computer. It derives time pulses from the 60-cycle computer power supply. These pulses are then used to produce a timing mark every 16.67 milliseconds or optionally every 8.33 milliseconds. The Real-Time Clock can also accept timing marks from a customer-supplied input, thereby allowing time measurement to any required resolution for special applications. These timing marks are supplied at standard SDS logic levels to the computer's RTC circuitry.

The timing marks are then used by the computer and its interrupt system to provide either an elapsed-time counter or a continuously incrementing time counter depending on the needs of the customer. The RTC will operate in either mode depending only on the computer's stored program.

Two pairs of locations of priority interrupts are provided with the RTC. These are as follows:

Location	Type	Computer	Description
164	Normal	92	CLOCK SYNC
166	Single Instruction	92	CLOCK PULSE

The Clock Pulse and Clock Sync interrupts function together to provide elapsed-time, event-counter or time-of-day clocks.

The Clock Pulse interrupt is a single-instruction interrupt. (Note: See Single Instruction Interrupts in Section III.) An MPO instruction is usually placed in the Clock Pulse interrupt location. When MPO is used as a single-instruction interrupt subroutine, it causes the contents of the effective address to be incremented by one but it does not alter the current content of the flag. Furthermore, if the new (incremented) contents of the effective address is 0000, a Clock Sync interrupt is generated. The Clock Sync interrupt can be generated in no other way.

ELAPSED-TIME CLOCK

The elapsed-time clock times the length of a program or subroutine, or initiates or discontinues processing at program-determined time intervals. An arbitrary memory location is reserved as a counter. When initialized, this cell contains the 2's complement of the number of time intervals to be counted. The Clock Pulse interrupt location contains an MPO instruction.

Each Clock Pulse interrupt results in incrementing the clock count by one. When the count is finished, an interrupt to the Clock Sync location occurs. A supervisory or other appropriate control program can then be entered to perform the customer-desired operation.

CONTINUOUSLY INCREMENTING CLOCK

The continuously incrementing clock maintains "time-of-day" for the computer. Two memory locations serve to count the timing marks. In this case, the Clock Pulse is used to increment the least significant twelve bits of the count. (The Clock Pulse interrupt location contains an MPO instruction.) The Clock Sync is used to increment the most significant twelve bits of the count. (The Clock Sync interrupt subroutine includes an MPO instruction.) A simple, straightforward subroutine can be entered to reconstruct the exact time-of-day from this twenty-four bit count.

ARM/DISARM

The Clock Pulse interrupt can be armed and disarmed with these instructions:

EOM Effective Address	Action
20200	Disarm Clock Pulse Interrupt
20100	Arm Clock Pulse Interrupt

The Clock Sync interrupt is always armed.

AUTOMATIC POWER FAIL-SAFE SYSTEM

The computer core memory holds its information with all power removed, but information in the computer registers is destroyed by loss of power. Upon failure of main power to the computer, this system provides that the contents of all registers and other volatile information are automatically stored in core memory; also, further writing into core storage is inhibited during the decay period of the computer dc power supply outputs. Erroneous memory control is prevented during power-off and power-on operations. Power-off/-on interrupt routines permit proper resumption of a program, automatically, after power is restored.

The system consists of relay-controlled, ac power-sensing and memory-sequencing circuitry, two high-priority interrupt channels, and a "shut-down/start-up" programming sequence.

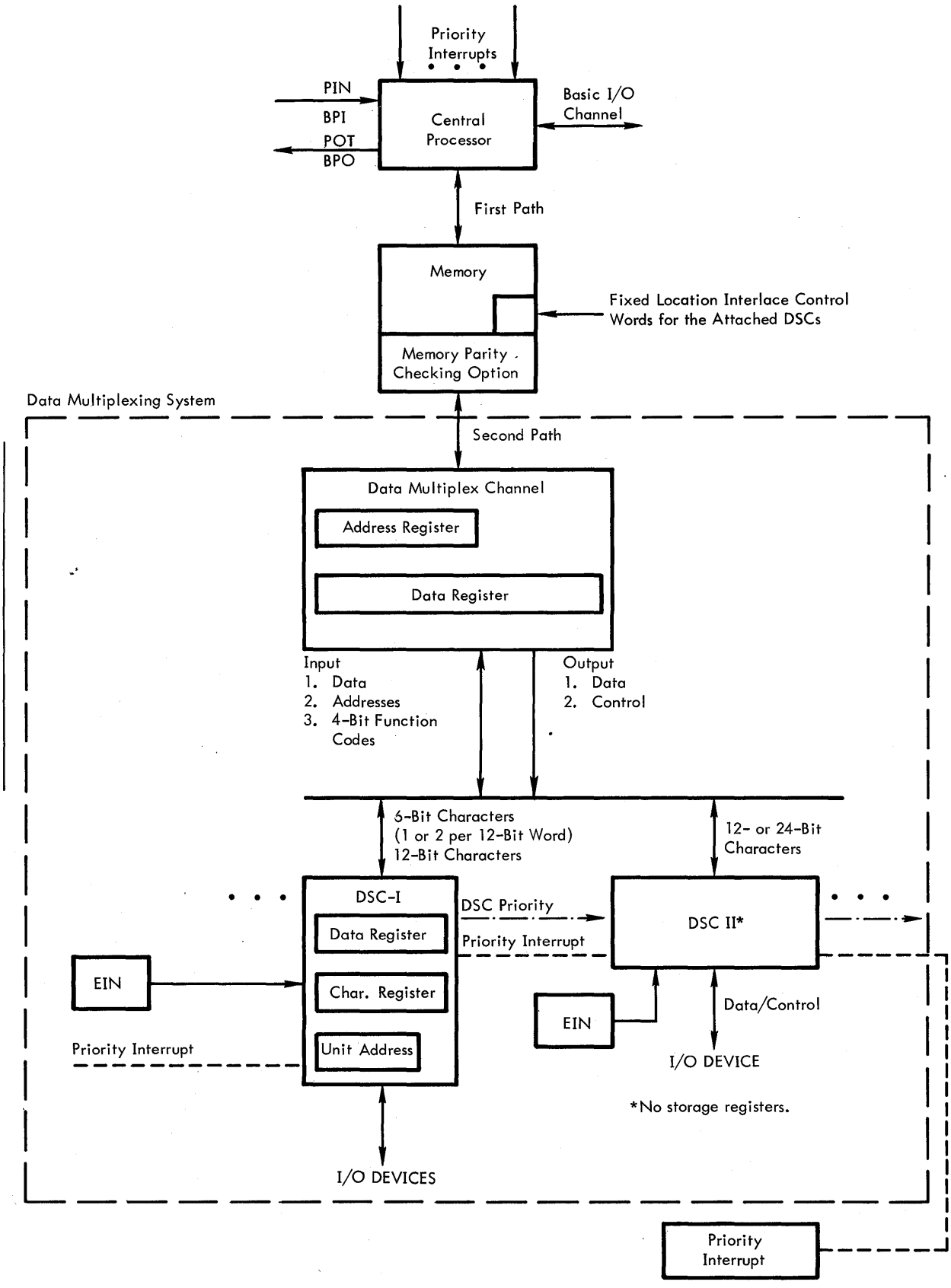
The Sense External Signal (SES) instruction is an aid in programming this option. Its effective address is 24000. If the OFF interrupt (152) has just occurred, this SES sets the Flag.

DATA MULTIPLEXING SYSTEM

INTRODUCTION

The standard I/O systems provided with the SDS 92 Computer provide for operation with all standard SDS peripheral equipments and for high-performance special devices. The Data Multiplexing System provides an alternate I/O system that is of particular use in dealing with multiple sources of data and for systems which may have data rates from low to very high.

The SDS 92 Computer has essentially two major paths along which I/O data can flow to and from memory. The first path is the same that is used by the main frame itself. The



SDS 92 Computer Configuration

PIN/POT operations use the first path. The basic I/O channel also uses this path. In addition to this path, which is primarily under the control of the main frame, there is an optional second path that is completely under the control of the units attached to it. The second path has priority over the first for access to memory. This path is made available with the installation of a Data Multiplexing System.

The Data Multiplexing System (DMS) consists of a Data Multiplex Channel (DMC) and one or more Data Subchannels (DSC). A maximum of 64 subchannels are allowed. Transmission between a DSC and computer memory is controlled by two interlace control word-pairs unique to the DSC and wired into fixed, adjacent locations in memory. During a transmission, the controlling DMC uses these two word-pairs for control of address and record length. Four DSCs (one DSC-I and three DSC-IIs) could be placed in a system as follows. The DSCs are numbered from 100, 104, ... to 114_8. Control word quads associated with the DSCs are numbered accordingly: 100-103 for DSC-100, etc.

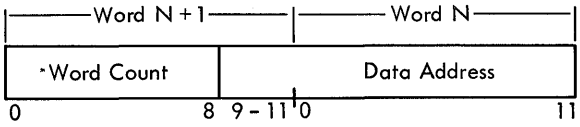
DATA MULTIPLEX CHANNEL (DMC)

The Data Multiplex Channel is a basic unit for the Data Multiplexing System. It connects to the SDS 92 Computer via the second path to memory. The DMC contains a 13-bit Data Register, 15-bit Address Register, and control logic to enable the DMS to perform a variety of functions. The data and address are connected to memory when a transfer of information is imminent. Program control required for input/output operates directly on the individual Data Subchannel (DSC), not on the DMC.

When external data addresses are provided to the DMC, the DMC transmissions require one cycle for each 12-bit word transmitted and two cycles for each 24-bit double word transmitted.

The DMC has an internal interlace feature. This feature allows subchannels to specify the addresses of word pairs in memory where the data address and count are located. When operating with internal interlace, the subchannel supplies the address of its associated interlace control words instead of the actual data address. The DMC accesses the interlace word pair, increments the address portion, decrements the word count, restores modified words, and then accepts data from, or transmits data to, the requesting subchannel. The DMC also supplies a signal to the subchannel, if the decremented word count is zero.

The format of the internal interlace word pair is:



The 9-bit word count permits block lengths to 512 words. Transmissions using internal interlace require five cycles, if the required transmission is for a 12-bit word, and six cycles, if the required transmission is for a 24-bit double word.

The DMC provides for automatic memory incrementing. The counting capability of the DMC Data Register permits an externally specified memory word to be incremented. When such a memory increment operation is to be performed, the subchannel signals the DMC with a special increment line and supplies the address. For memory increments, the DMC accesses memory, increments it, and then restores the word. If a memory increment operation results in an all-zero word (or double word), the DMC signals the subchannel. The zero signal may then be used to interrupt the program. Memory increments require two cycles.

Control of the various DMC functions is achieved by four Function Code lines from the subchannels. The DMC, in conjunction with the main frame Memory Parity Checking Option, insure the integrity of data transmissions. Words read from memory are checked for parity; parity is generated for words stored in memory; words received by the DMC are checked for proper parity; a parity error signal is generated by the DMC and sent to the subchannel when an input parity discrepancy is detected.

DATA SUBCHANNELS (DSC-N)

A number of subchannels can be attached to the DMC. The two described below are standard subchannels. Subchannels can control and generate program interrupts, but do not include the interrupt levels themselves. The signals must be routed to optional interrupt levels.

The subchannels use a priority scheme to determine which may transmit to the DMC at any given time. Up to 64 DSCs may be connected to a DMC. A DSC may use the internal interlace feature of the DMC to control its transmission, or it may be equipped with an External Interlace (EIN).

A DSC using internal interlace has two word pairs assigned to it. These two word pairs are located in contiguous memory locations and are fixed for a given subchannel. The program may select either the even word pair or odd word pair location. If the even word pair location is selected, the subchannel will automatically switch to the odd word pair location when the count field of the even word pair word is zero. The program can also select whether the subchannel switches back to the even word pair when the count field of the odd word pair is zero. The subchannel generates an interrupt signal when the count field of either word pair reaches zero. Transmission termination occurs when the odd word pair's count equals zero, if the subchannel does not switch back to the even word pair.

The two word pair internal interlace allows a subchannel to handle continuous data by alternately working from one memory area or another. By allowing the subchannel to switch automatically from one interlace word pair to the other, the program is relieved of the necessity for making real-time responses to the zero count condition. Using first the even pair then the odd pair interlace words allows a maximum transmission of 1024 words or double words.

CHARACTER SUBCHANNEL (DSC-I)

The DSC-I contains a 12-bit data register that can assemble and disassemble two 6-bit characters, and transmit one or two 6-bit characters or one 12-bit character. (DSC-I has a unit address

register.) It checks and generates the parity of characters to enable it to couple with standard SDS peripheral equipment.

The subchannel may operate with either internal or external interlace. It has one mode of output and two modes of input. During output, it transmits until the odd internal interlace word pair count is zero and then terminates, if interlace cycling is not requested. The output may also be terminated if the device sends an END signal to the channel. This END signal may cause the DSC-I to generate an interrupt to the program.

Input, like output, may always be terminated due to an external END signal. The program can also specify that the DSC terminates and disconnects on zero count, or disconnects only on the END signal. In either case, however, all transmission to memory is terminated after the odd interlace count reaches zero, if interlace cycling is not requested.

FULL WORD SUBCHANNEL (DSC-II)

The DSC-II is a general-purpose subchannel, designed to allow any device to be connected to it. It contains no storage for data. Depending on the Function Code provided to the DMC, the DSC-II will permit 12- or 24-bit (plus parity) transmission between the DMC and external devices connected to the DSC-II. The external device must be capable of holding the data during the transmission to/from the DMC. (An A-to-D converter would have such capability.) Like the DSC-I, the DSC-II can operate with either internal or external interlace. Its operation in this respect is identical to the DSC-I. The DSC-II also contains control logic to facilitate memory increment operations in conjunction with the DMC.

EXTERNAL INTERLACE (EIN)

The External Interlace can be attached to the DSC to control the transmission of its data to/from memory. The EIN consists of a 15-bit address register and a 9-bit count register. These registers are loaded automatically when the subchannel is activated, the information coming from the internal interlace memory locations. Once the EIN is set up, it will control the transmission of the DSC at a maximum rate of one word per memory cycle. After each word is transmitted, the EIN increments its address register and decrements its count. When the count equals zero, the EIN signals the DSC, which can then generate a program interrupt and/or notify the external device. Transmission normally terminates on zero count. Sequencing of interlace words is identical to the sequence of operations performed for internal interlace, except that only four memory cycles are used for interlace word processing. The first is to access the interlace word pair initially; the second is to restore the interlace word pair when the count reaches zero.

MEMORY PARITY INTERRUPTS

SDS computers incorporate an extensive memory parity checking system. The inclusion of parity generation and checking circuitry assures the integrity of all data and instructions transferred among the memory, the central processing unit, and input/output channels.

In normal operation a switch on the computer console specifies the action to be performed by the computer when a memory parity error is detected. Two actions are available: the computer halts with the parity indicator lighted; or the computer ignores the parity error and proceeds with the program.

In many real-time applications it is desirable to keep the computer running when a parity error is detected. Also, the program must be notified of the error without stopping computation.

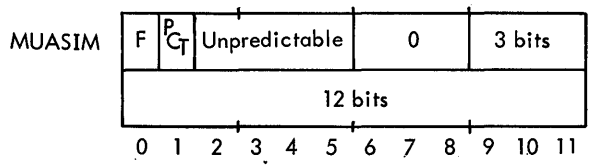
An optional feature provides this capability by means of two levels of armed interrupts. One interrupt level is associated with the central processor and the Time-Multiplexed Communication Channels; the other interrupt level with the Direct Access Communication Channels and the Data Multiplexing System. Memory parity errors detected from these two sources produce a priority interrupt associated with the cause. The processing routine associated with the interrupt can then take appropriate action, such as re-initiate the failed operation, notify the operator, or enter a diagnostic routine. Such action allows memory parity errors to be recognized and handled properly without hindering the computer's performance of real-time or on-line calculations.

TRAPPING RETURN SUBROUTINE EXAMPLE

The following code determines how many cells (one or two) the trapped instruction used: it then increments the subroutine entry accordingly to provide the proper return address.

Assume (1) the trap instruction is a multiply simulator at location 124, and (2) the branch in location 124 is BRM MUASIM. Assume also that MUA hardware is not present in the machine.

Executing an instruction containing the MUA operation code causes the BRM MUASIM to be executed. The marked place has the form:



which marks the Flag and PCT bits in 0, 1, zeros in bits 6 through 8, and the trap instruction address in bits 9, 10, 11 to MUASIM and in bits 0 through 11 in MUASIM+1.

The subroutine return routine follows:

```
MUASIM      DATA
PLUSONE     DATA
            LDB     *MUASIM     Load trapped instruction.
            LDA     =1          Load an incrementer.
            COB     =040
            BFT     TRUE        Branch if bit 6 is reset.
            COB     =037
            BRU     ACK         F = 1 if address was an immediate address (two words); F = 0 if it
                                was direct single precision.
TRUE        COB     =010
            BFT     ACK         Branch if bit 8 is reset (i.e., if F is set to 1 which implies two
                                word-full address with no index).
            COB     =020        F = 0 if indirect address, single precision; F = 1 if two word
                                indexed
ACK         ACA     =0          Add 1 to A if two word (add 0 if one word).
            MPA     MUASIM+1    Add 1 or 2 to the subroutine entry.
            MPF     MUASIM      This is for address overflow correction.
```

SDS 92 MEMORY ALLOCATION

	0000	Unassigned
	0001 - 0037	Scratch Pad
	0040 - 0077	Unassigned
	0100 - 0117	DSC Interlace Control Word Pairs
	0120	Trap 12
	0122	Trap 52
	0124	Trap 13
	0126	Trap 53
	0130	Trap 10
	0132	Trap 50
	0134	Trap 11
	0136	Trap 51
	0140	Trap 14
	0142	Trap 54
	0144	Trap 15
	0146	Trap 55
Is	0150	Interrupt, POWER ON (always armed)
Is	0152	Interrupt, POWER OFF (always armed)
Is	0154	Interrupt, MAIN FRAME PARITY (armed via console switch)
Is	0156	Interrupt, DATA MULTIPLEXING SYSTEM PARITY (armed via console switch)
Is	0160	Unassigned
Is	0162	Unassigned
Is	0164	Interrupt, CLOCK SYNC (always armed)
Is	0166	Interrupt, CLOCK PULSE (arm furnished, I_i type)
Is	0170	Interrupt, I1 (arm furnished)
Is	0172	Interrupt, I2 (arm furnished)
Ir	0174	Unassigned
Ir	0176	Unassigned
Ir	0200 - 1177	System Interrupts (up to 256 levels; any may be of I_i type if desired)

I_i → Single Instruction Interrupt

Ir → Interrupt system must be enabled before interrupt goes active.

Is → Interrupt always proceeds from Waiting to Active

SDS 92 INSTRUCTION LIST — FUNCTIONAL CATEGORIES

Mnemonic	Instruction Code	Name	Function	Timing*
LOAD/STORE				
LDA	64	LOAD A	(M) → A	2
LDB	24	LOAD B	(M) → B	2
STA	44	STORE A	(A) → M	2
STB	04	STORE B	(B) → M	2
XMA	74	EXCHANGE M AND A	(A) ↔ (M)	3
XMB	34	EXCHANGE M AND B	(B) ↔ (M)	3
FLAG				
XMF	17	EXCHANGE M AND F	$(M)_0$ ↔ F	3
LDF	57	LOAD F	$(M)_0$ → F	3
SFT	0044	SET FLAG TRUE	1 → F	3,4
SFF	0042	SET FLAG FALSE	0 → F	3,4
INF	0046	INVERT FLAG	If (F)=1, 0 → F; if (F)=0, 1 → F	3,4
ARITHMETIC				
ADA	62	ADD TO A	(A)+(M) → A; Carry → F	2
ADB	22	ADD TO B	(B)+(M) → B; Carry → F	2
ACA	63	ADD WITH CARRY TO A	(A)+(M)+F → A; Carry → F	2
ACB	23	ADD WITH CARRY TO B	(B)+(M)+F → B; Carry → F	2
SUA	60	SUBTRACT TO A	(A)-(M) → A; Carry → F	2
SUB	20	SUBTRACT TO B	(B)-(M) → B; Carry → F	2
SCA	61	SUBTRACT WITH CARRY TO A	(A)-(M)-F → A; Carry → F	2
SCB	21	SUBTRACT WITH CARRY TO B	(B)-(M)-F → B; Carry → F	2
MPA	76	MEMORY PLUS A TO MEMORY	(M)+(A) → M; Carry → F	3
MPB	36	MEMORY PLUS B TO MEMORY	(M)+(B) → M; Carry → F	3
MPO	16	MEMORY PLUS ONE TO MEMORY	(M)+1 → M; Carry → F	3
MPF	56	MEMORY PLUS FLAG TO MEMORY	(M)+(F) → M; Carry → F	3
MUA	13	MULTIPLY A (Optional)	(A)×(M) → AB	5
MUB	53	MULTIPLY B (Optional)	(B)×(M) → AB	5
DVA	52	DIVIDE AB (Optional)	(AB)÷(M) → B; R → A	13
DVB	12	DIVIDE BA (Optional)	(BA)÷(M) → B; R → A	13
LOGICAL				
ANA	65	AND TO A	(A) and (M) → A	2
ANB	25	AND TO B	(B) and (M) → B	2
ORA	67	OR TO A	(A) or (M) → A	2
ORB	27	OR TO B	(B) or (M) → B	2
EOA	66	EXCLUSIVE OR TO A	$(M)(\overline{A})$ or $(\overline{M})(A)$ → A	2

*See page 2-1 for interpretation and use of the Timing column.

Mnemonic	Instruction Code	Name	Function	Timing
LOGICAL (continued)				
EOB	26	EXCLUSIVE OR TO B	$(M)(\overline{B})$ or $(\overline{M})(B) \longrightarrow B$	2
MAA	75	MEMORY AND A TO MEMORY	(M) and (A) $\longrightarrow$ M	3
MAB	35	MEMORY AND B TO MEMORY	(M) and (B) $\longrightarrow$ M	3
COMPARISON				
COA	45	COMPARE ONES WITH A	If (A)(M)=0, set F; if (A)(M)$\neq$0, reset F	2
COB	05	COMPARE ONES WITH B	If (B)(M)=0, set F; if (B)(M)$\neq$0, reset F	2
CMA	47	COMPARE MAGNITUDE OF M WITH A	If (A)$\geqslant$(M), set F; if (A)<(M), reset F	2
CMB	07	COMPARE MAGNITUDE OF M WITH B	If (B)$\geqslant$(M), set F; if (B)<(M), reset F	2
CEA	46	COMPARE M EQUAL TO A	If (M)$\neq$(A), set F; if (M)=(A), reset F	2
CEB	06	COMPARE M EQUAL TO B	If (M)$\neq$(B), set F; if (M)=(B), reset F	2
BRANCH				
BRU	73	BRANCH UNCONDITIONALLY	M $\longrightarrow$ P	1
BRC	32	BRANCH, CLEAR INTERRUPT, AND LOAD FLAG	M $\longrightarrow$ P; clear Interrupt (see page 2-4)	3
BRL	33	BRANCH AND LOAD FLAG	M $\longrightarrow$ P (see page 2-4)	1
BFF	31	BRANCH ON FLAG FALSE	If F = 0, M $\longrightarrow$ P; if F = 1, take next instruction	1 2
BFT	71	BRANCH ON FLAG TRUE	If F = 1, M $\longrightarrow$ P; if F = 0, take next instruction	1 2

BDA	70	BRANCH ON DECREMENTING A	(A) - 1 $\longrightarrow$ A
			If (A) $\neq 7777_8$, M $\longrightarrow$ P;
			If (A) = 7777_8, take next instruction
				
				1
				2

BAX	30	BRANCH AND EXCHANGE A AND B	(A) $\longleftrightarrow$ (B); (M) $\longrightarrow$ P	1
BRM	77	BRANCH AND MARK PLACE	(P) $\longrightarrow$ M, M+1; M+2 $\longrightarrow$ P; (F) $\longrightarrow M_0$; (PCT) $\longrightarrow M_1$	3
BMC	37	BRANCH, MARK PLACE, AND CLEAR FLAG	(See page 2-5)	3
SHIFT				
CYA	42*	CYCLE A	A cycled left N places	3-7
CYB	02*	CYCLE B	B cycled left N places	3-7
CFA	43*	CYCLE FLAG AND A	F, A cycled left N places	3-7
CFB	03*	CYCLE FLAG AND B	F, B cycled left N places	3-7
CYD	02/42* **	CYCLE DOUBLE	A, B cycled left N places	3-7
CFD	43*	CYCLE FLAG AND DOUBLE	A, B, F cycled left N places	3-7
CFI	03*	CYCLE FLAG AND DOUBLE INVERSE	B, A, F cycled left N places	3-7
CONTROL				
EXU	72	EXECUTE	Instruction M is performed, P unchanged	1
HLT	0041/00000000**	HALT	Halts computation	3,4

*See page 2-5 for indication of the instruction structure and code redundancy.

**A slash (/) indicates that either instruction code can be used to perform the same operation.

SDS 92 INSTRUCTION LIST – FUNCTIONAL CATEGORIES (continued)

Mnemonic	Instruction Code	Name		Timing
TRAPPING				
SCT	0061	SET PROGRAM-CONTROLLED TRAP	1 → PCT	3, 4
RCT	0060	RESET PROGRAM-CONTROLLED TRAP	0 → PCT	3, 4
TCT	0160	TEST PROGRAM-CONTROLLED TRAP	If PCT = 0, 0 → F; if PCT = 1, 1 → F	3, 4
BREAKPOINT TESTS				
BPT 1	0144	BREAKPOINT NO. 1 TEST	Test Breakpoint Switch	3, 4
BPT 2	0145	BREAKPOINT NO. 2 TEST	Test Breakpoint Switch	3, 4
BPT 3	0146	BREAKPOINT NO. 3 TEST	Test Breakpoint Switch	3, 4
BPT 4	0147	BREAKPOINT NO. 4 TEST	Test Breakpoint Switch	3, 4
INTERRUPTS				
EIR	0051	ENABLE INTERRUPT		3, 4
DIR	0050	DISABLE INTERRUPT		3, 4
IET	0150	INTERRUPT ENABLED TEST; SET FLAG IF INTERRUPT SYSTEM ENABLED		3, 4
AIR	00020001	ARM INTERRUPTS		3, 4
CHANNEL CONTROL AND TESTS				
DSC	00000100	DISCONNECT CHANNEL		3, 4
TOP	00012100	TERMINATE OUTPUT ON CHANNEL		3, 4
TIP	00012100	TERMINATE INPUT ON CHANNEL		3, 4
ALC	00050100	ALERT CHANNEL INTERLACE		3, 4
ASC	00010500	ALERT TO STORE INTERLACE COUNT		3, 4
CAT	01004100	CHANNEL ACTIVE TEST; SET FLAG IF NOT ACTIVE		3, 4
CET	01001100	CHANNEL ERROR TEST; SET FLAG IF ERROR		3, 4
INPUT/OUTPUT				
WIN	15	WORD IN	(Channel) → M	5 + wait
RIN	55	RECORD IN	(Channel$_{N\ words}$) → M	3 + 2N + wait
WOT	11	WORD OUT	(M) → Channel	4 + wait
ROT	51	RECORD OUT	(M$_{N\ words}$) → Channel	2 + 2N + wait
PIN	14	PARALLEL INPUT	(Unit M) → M in parallel	5 + wait, and 5, 6 + wait
POT	10	PARALLEL OUTPUT	(M) → Unit M in parallel	4 + wait, and 4, 5 + wait
BPI	54	BLOCK PARALLEL INPUT	(Unit M) → M in parallel, N sequential locations	4 + N + wait, and 3, 4 + 2N + wait
BPO	50	BLOCK PARALLEL OUTPUT	(M) → Unit M in parallel, N sequential locations	3 + N + wait, and 2, 3 + 2N + wait
EOM	00(40)*	ENERGIZE OUTPUT M	3.5 μsec pulse to points addressed	3, 4
SES	01(41)*	SENSE EXTERNAL SIGNAL	If Signal = 1, set Flag Bit; if Signal = 0, reset Flag Bit	3, 4

*Codes EOM 40 and SES 41 are reserved for use in special system applications.

SDS 92 INSTRUCTION LIST — NUMERICAL ORDER

Instruction Code	Mnemonic	Name	Page Reference
00(40)*	EOM	ENERGIZE OUTPUT M	3-5, 3-6, 3-9
00000000/0041**	HLT	HALT	2-6
00000100	DSC	DISCONNECT CHANNEL	3-6
00010500	ASC	ALERT TO STORE INTERLACE COUNT	3-6
00012100	TIP	TERMINATE INPUT ON CHANNEL	3-6
00012100	TOP	TERMINATE OUTPUT ON CHANNEL	3-6
00020001	AIR	ARM INTERRUPTS	3-12
00050100	ALC	ALERT CHANNEL INTERLACE	3-7
0042	SFF	SET FLAG FALSE	2-2
0044	SFT	SET FLAG TRUE	2-1
0046	INF	INVERT FLAG	2-2
0050	DIR	DISABLE INTERRUPT	3-11
0051	EIR	ENABLE INTERRUPT	3-11
0060	RCT	RESET PROGRAM-CONTROLLED TRAP	2-7
0061	SCT	SET PROGRAM-CONTROLLED TRAP	2-7
01(41)*	SES	SENSE EXTERNAL SIGNAL	3-6, 3-7, 3-9
01001100	CET	CHANNEL ERROR TEST; SET FLAG IF ERROR	3-7
01004100	CAT	CHANNEL ACTIVE TEST; SET FLAG IF NOT ACTIVE	3-7
0144	BPT 1	BREAKPOINT NO. 1 TEST	2-7
0145	BPT 2	BREAKPOINT NO. 2 TEST	2-7
0146	BPT 3	BREAKPOINT NO. 3 TEST	2-7
0147	BPT 4	BREAKPOINT NO. 4 TEST	2-7
0150	IET	INTERRUPT ENABLED TEST; SET FLAG IF INTERRUPT SYSTEM ENABLED	3-11
0160	TCT	TEST PROGRAM-CONTROLLED TRAP	2-7
02***	CYB	CYCLE B	2-6
02/42** ***	CYD	CYCLE DOUBLE	2-6
03***	CFB	CYCLE FLAG AND B	2-6
03***	CFI	CYCLE FLAG AND DOUBLE INVERSE	2-6
04	STB	STORE B	2-1
05	COB	COMPARE ONES WITH B	2-4
06	CEB	COMPARE M EQUAL TO B	2-4
07	CMB	COMPARE MAGNITUDE OF M WITH B	2-4
10	POT	PARALLEL OUTPUT	3-7, 3-8, 3-9, 3-12
11	WOT	WORD OUT	3-2
12	DVB	DIVIDE BA (Optional)	2-3
13	MUA	MULTIPLY A (Optional)	2-3

*Codes EOM 40 and SES 41 are reserved for use in special system applications.

**A slash (/) indicates that either instruction code can be used to perform the same operation.

***See page 2-5 for indication of the instruction structure and code redundancy.

Instruction Code	Mnemonic	Name	Page Reference
14	PIN	PARALLEL INPUT	3-8, 3-9
15	WIN	WORD IN	3-2
16	MPO	MEMORY PLUS ONE TO MEMORY	2-2
17	XMF	EXCHANGE M AND F	2-1
20	SUB	SUBTRACT TO B	2-2
21	SCB	SUBTRACT WITH CARRY TO B	2-2
22	ADB	ADD TO B	2-2
23	ACB	ADD WITH CARRY TO B	2-2
24	LDB	LOAD B	2-1
25	ANB	AND TO B	2-3
26	EOB	EXCLUSIVE OR TO B	2-3
27	ORB	OR TO B	2-3
30	BAX	BRANCH AND EXCHANGE A AND B	2-5
31	BFF	BRANCH ON FLAG FALSE	2-4
32	BRC	BRANCH, CLEAR INTERRUPT, AND LOAD FLAG	2-4
33	BRL	BRANCH AND LOAD FLAG	2-4
34	XMB	EXCHANGE M AND B	2-1
35	MAB	MEMORY AND B TO MEMORY	2-4
36	MPB	MEMORY PLUS B TO MEMORY	2-2
37	BMC	BRANCH, MARK PLACE, AND CLEAR FLAG	2-5
42*	CYA	CYCLE A	2-6
43*	CFA	CYCLE FLAG AND A	2-6
43*	CFD	CYCLE FLAG AND DOUBLE	2-6
44	STA	STORE A	2-1
45	COA	COMPARE ONES WITH A	2-4
46	CEA	COMPARE M EQUAL TO A	2-4
47	CMA	COMPARE MAGNITUDE OF M WITH B	2-4
50	BPO	BLOCK PARALLEL OUTPUT	3-8, 3-9
51	ROT	RECORD OUT	3-2
52	DVA	DIVIDE AB (Optional)	2-3
53	MUB	MULTIPLY B (Optional)	2-3
54	BPI	BLOCK PARALLEL INPUT	3-8, 3-9
55	RIN	RECORD IN	3-3
56	MPF	MEMORY PLUS FLAG TO MEMORY	2-3
57	LDF	LOAD F	2-1
60	SUA	SUBTRACT TO A	2-2
61	SCA	SUBTRACT WITH CARRY TO A	2-2
62	ADA	ADD TO A	2-2

*See page 2-5 for indication of the instruction structure and code redundancy.

SDS 92 INSTRUCTION LIST – NUMERICAL ORDER (continued)

Instruction Code	Mnemonic	Name	Page Reference
63	ACA	ADD WITH CARRY TO A	2-2
64	LDA	LOAD A	2-1
65	ANA	AND TO A	2-3
66	EOA	EXCLUSIVE OR TO A	2-3
67	ORA	OR TO A	2-3
70	BDA	BRANCH ON DECREMENTING A	2-5
71	BFT	BRANCH ON FLAG TRUE	2-5
72	EXU	EXECUTE	2-6
73	BRU	BRANCH UNCONDITIONALLY	2-4
74	XMA	EXCHANGE M AND A	2-1
75	MAA	MEMORY AND A TO MEMORY	2-4
76	MPA	MEMORY PLUS A TO MEMORY	2-2
77	BRM	BRANCH AND MARK PLACE	2-5

SDS 92 INSTRUCTION LIST — ALPHABETICAL ORDER

Mnemonic	Instruction Code	Name	Page Reference
ACA	63	ADD WITH CARRY TO A	2-2
ACB	23	ADD WITH CARRY TO B	2-2
ADA	62	ADD TO A	2-2
ADB	22	ADD TO B	2-2
AIR	00020001	ARM INTERRUPTS	3-12
ALC	00050100	ALERT CHANNEL INTERLACE	3-7
ANA	65	AND TO A	2-3
ANB	25	AND TO B	2-3
ASC	00010500	ALERT TO STORE INTERLACE COUNT	3-6
BAX	30	BRANCH ON DECREMENTING A	2-5
BDA	70	BRANCH AND DECREMENTING A	2-5
BFF	31	BRANCH ON FLAG FALSE	2-4
BFT	71	BRANCH ON FLAG TRUE	2-5
BMC	37	BRANCH, MARK PLACE, AND CLEAR FLAG	2-5
BPI	54	BLOCK PARALLEL INPUT	3-8, 3-9
BPO	50	BLOCK PARALLEL OUTPUT	3-8, 3-9
BPT 1	0144	BREAKPOINT NO. 1 TEST	2-7
BPT 2	0145	BREAKPOINT NO. 2 TEST	2-7
BPT 3	0146	BREAKPOINT NO. 3 TEST	2-7
BPT 4	0147	BREAKPOINT NO. 4 TEST	2-7
BRC	32	BRANCH, CLEAR INTERRUPT, AND LOAD FLAG	2-4
BRL	33	BRANCH AND LOAD FLAG	2-4
BRM	77	BRANCH AND MARK PLACE	2-5
BRU	73	BRANCH UNCONDITIONALLY	2-4
CAT	01004100	CHANNEL ACTIVE TEST; SET FLAG IF NOT ACTIVE	3-7
CEA	46	COMPARE M EQUAL TO A	2-4
CEB	06	COMPARE M EQUAL TO B	2-4
CET	01001100	CHANNEL ERROR TEST; SET FLAG IF ERROR	3-7
CFA	43*	CYCLE FLAG AND A	2-6
CFB	03*	CYCLE FLAG AND B	2-6
CFD	43*	CYCLE FLAG AND DOUBLE	2-6
CFI	03*	CYCLE FLAG AND DOUBLE INVERSE	2-6
CMA	47	COMPARE MAGNITUDE OF M WITH A	2-4
CMB	07	COMPARE MAGNITUDE OF M WITH B	2-4
COA	45	COMPARE ONES WITH A	2-4
COB	05	COMPARE ONES WITH B	2-4
CYA	42*	CYCLE A	2-6
CYB	02*	CYCLE B	2-6

*See page 2-5 for indication of the instruction structure and code redundancy.

Mnemonic	Instruction Code	Name	Page Reference
CYD	02/42* **	CYCLE DOUBLE	2-6
DIR	0050	DISABLE INTERRUPT	3-11
DSC	00000100	DISCONNECT CHANNEL	3-6
DVA	52	DIVIDE AB (Optional)	2-3
DVB	12	DIVIDE BA (Optional)	2-3
EIR	0051	ENABLE INTERRUPT	3-11
EOA	66	EXCLUSIVE OR TO A	2-3
EOB	26	EXCLUSIVE OR TO B	2-3
EOM	00(40)***	ENERGIZE OUTPUT M	3-5, 3-6, 3-9
EXU	72	EXECUTE	2-6
HLT	00000000/0041**	HALT	2-6
IET	0150	INTERRUPT ENABLED TEST; SET FLAG IF INTERRUPT SYSTEM ENABLED	3-11
INF	0046	INVERT FLAG	2-2
LDA	64	LOAD A	2-1
LDB	24	LOAD B	2-1
LDF	57	LOAD F	2-1
MAA	75	MEMORY AND A TO MEMORY	2-4
MAB	35	MEMORY AND B TO MEMORY	2-4
MPA	76	MEMORY PLUS A TO MEMORY	2-2
MPB	36	MEMORY PLUS B TO MEMORY	2-2
MPF	56	MEMORY PLUS FLAG TO MEMORY	2-3
MPO	16	MEMORY PLUS ONE TO MEMORY	2-2
MUA	13	MULTIPLY A (Optional)	2-3
MUB	53	MULTIPLY B (Optional)	2-3
ORA	67	OR TO A	2-3
ORB	27	OR TO B	2-3
PIN	14	PARALLEL INPUT	3-8, 3-9
POT	10	PARALLEL OUTPUT	3-7, 3-8, 3-9, 3-12
RCT	0060	RESET PROGRAM-CONTROLLED TRAP	2-7
RIN	55	RECORD IN	3-3
ROT	51	RECORD OUT	3-2
SCA	61	SUBTRACT WITH CARRY TO A	2-2
SCB	21	SUBTRACT WITH CARRY TO B	2-2
SCT	0061	SET PROGRAM-CONTROLLED TRAP	2-7
SES	01(41)***	SENSE EXTERNAL SIGNAL	3-6, 3-7, 3-9
SFF	0042	SET FLAG FALSE	2-2
SFT	0044	SET FLAG TRUE	2-1

*See page 2-5 for indication of the instruction structure and code redundancy.

**A slash (/) indicates that either instruction code can be used to perform the same operation.

***Codes EOM 40 and SES 41 are reserved for use in special system applications.

Mnemonic	Instruction Code	Name	Page Reference
STA	44	STORE A	2-1
STB	04	STORE B	2-1
SUA	60	SUBTRACT TO A	2-2
SUB	20	SUBTRACT TO B	2-2
TCT	0160	TEST PROGRAM-CONTROLLED TRAP	2-7
TIP	00012100	TERMINATE INPUT ON CHANNEL	3-6
TOP	00012100	TERMINATE OUTPUT ON CHANNEL	3-6
WIN	15	WORD IN	3-2
WOT	11	WORD OUT	3-2
XMA	74	EXCHANGE M AND A	2-1
XMB	34	EXCHANGE M AND B	2-1
XMF	17	EXCHANGE M AND F	2-1

SDS 92 INPUT/OUTPUT INSTRUCTIONS

Buffer Instructions and Tests

BUFFER CONTROL

Mnemonic		Octal Code	Name
EOM	A,T	00 or 40	Energize Output M
DSC		00000100	Disconnect Channel
TOP		00012100	Terminate Output on Channel
TIP		00012100	Terminate Input on Channel
ASC		00010500	Alert to Store Interlace Count
ALC		00050100	Alert Channel Interlace

PARALLEL INPUT/OUTPUT

Mnemonic		Octal Code	Name
POT	A,T	10	Parallel Output
BPO	A,T	50	Block Parallel Output
PIN	A,T	14	Parallel Input
BPI	A,T	54	Block Parallel Input

BUFFER TESTS

Mnemonic		Octal Code	Name
SES	A,T	01 or 41	Sense External Signal
CAT		01004100	Channel Active Test
CET		01001100	Channel Error Test

DATA TRANSFER

Mnemonic		Octal Code	Name
WOT	A,T	11	Word Out
ROT	A,T	51	Record Out
WIN	A,T	15	Word In
RIN	A,T	55	Record In

Peripheral Device Instructions and Tests

°TYPEWRITER

Mnemonic		Octal Code	Name
RKB	1,2	00002301	Read Keyboard
TYP	1,2	00002341	Write Typewriter

°PAPER TAPE

Mnemonic		Octal Code	Name
RPT	1,2	00002304	Read Paper Tape
PTL	1,2	00000344	Punch Paper Tape with Leader
PPT	1,2	00002344	Punch Paper Tape, No Leader

°CARD

Mnemonic		Octal Code	Name
CRT	1	01012106	Card Reader Ready Test
CFT	1	01011106	Card Reader EOF Test
RCD	1,2	00002306	Read Card Decimal (Hollerith)
RCB	1,2	00003306	Read Card Binary
CPT	1	01014146	Card Punch Ready Test
PCD	1,2	00002346	Punch Card Decimal (Hollerith)
PCB	1,2	00003346	Punch Card Binary

° LINE PRINTER

Mnemonic		Octal Code	Name
PRT	1	01012160	Printer Ready Test
EPT	1	01014160	End of Page Test
PFT	1	01011160	Printer Fault Test
POL	1	00010360	Printer Off-line

°LINE PRINTER (cont.)

Mnemonic		Octal Code	Name
PSC	1,n	0001n560	Printer Skip to Format Channel n
PSP	1,n	0001n760	Printer Space n Lines
PLP	1,2	00002360	Print Line Printer

MAGNETIC TAPE

Mnemonic		Octal Code	Name
TRT	n	0101051n	Tape Ready Test
FPT	n	0101411n	File Protect Test
BTT	n	0101211n	Beginning of Tape Test
ETT	n	0101111n	End of Tape Test
DT2	n	0101631n	Density Test, 200 BPI
DT5	n	0101671n	Density Test, 556 BPI
DT8	n	0101731n	Density Test, 800 BPI
TFT		01013710	Tape End-of-File Test
WTB	n,2	0000335n	Write Tape in Binary
WTD	n,2	0000235n	Write Tape in Decimal (BCD)
EFT	n,2	0000337n	Erase Forward Tape
ERT	n,2	0000737n	Erase Reverse Tape
RTB	n,2	0000331n	Read Tape in Binary
RTD	n,2	0000231n	Read Tape in Decimal (BCD)
SFB	n,2	0000333n	Scan Forward in Binary
SFD	n,2	0000233n	Scan Forward in Decimal (BCD)
SRB	n,2	0000733n	Scan Reverse in Binary
SRD	n,2	0000633n	Scan Reverse in Decimal (BCD)
REW	n	0001411n	Rewind
RTS	0	00014100	Convert Read to Scan

Legend

A = address; *A = indirect address; =A = immediate address; T = index tag; n = number (0-7)

°Mnemonics and Octal Codes are given for device number 1 in a two-character/word mode.

SCIENTIFIC DATA SYSTEMS 1649 Seventeenth Street • Santa Monica, California • Phone (213) UP 1-0960

SALES OFFICES

EASTERN

1301 Avenue of the Americas
New York City, New York
(212) 765-1230

69 Hickory Drive
Waltham, Massachusetts
(617) 899-4700

1145 Nineteenth Street, N. W.
Washington, D. C.
(202) 337-6838

One Bala Avenue Building
Bala-Cynwyd, Pennsylvania
(215) 667-4944

SOUTHERN

Holiday Office Center
3322 South Memorial Parkway
Huntsville, Alabama
(205) 881-5746

1325 North Atlantic Avenue
Cocoa Beach, Florida
(305) 784-1555

3334 Richmond Avenue
Houston, Texas
(713) 526-2693

6434 Maple Avenue
Dallas Texas
(214) 357-0451

MIDWEST

One Parkway Center
875 Greentree Road
Pittsburgh, Pennsylvania
(412) 921-3640

3105 Des Plaines Avenue
Des Plaines, Illinois
(312) 824-8147

WESTERN

2526 Broadway Avenue
Santa Monica, California
(213) 870-8562

Fountain Professional Building
9000 Menaul Blvd. N. E.
Albuquerque, New Mexico
(505) 298-8009

Sunnyvale Office Center
505 West Olive Avenue
Sunnyvale, California
(408) 736-9133

World Savings Building
1111 South Colorado Boulevard
Denver, Colorado
(303) 756-8505

Suite 100, Redwood Building
845 106th Street, N. E.
Bellevue, Washington
(206) 454-3991

1360 So. Anaheim Boulevard
Anaheim, California
(213) 865-5293 (F.X.)
(714) 774-0461 (Local)

FOREIGN REPRESENTATIVES

EUROPE

CECIS
14 Rue de la Baume
Paris 8, France

CANADA

INSTRONICS, Ltd.
P. O. Box 100
Stittsville
Ontario, Canada

JAPAN

F. Kanematsu & Co. Inc.
Central P. O. Box 141
New Kaijo Bldg.
Marunouchi
Tokyo, Japan

AUSTRALIA

RACAL Pty. Ltd.
5 Ridge Street
N. Sydney, NSW, Australia

90 05 05C