XDS

Xerox Data Systems

Reference Manual

XDS 940 Computer

XDS 940 INSTRUCTIONS

Mnemonic	Code	Name	Page
LOAD/STORE			
LDA	76	Load A	19
STA	35	Store A	19
LDB	75	Load B	19
STB	36	Store B	19
LDX	71	Load Index	19
STX	37	Store Index	19
EAX	77	Copy Effective Address into Index	20
XMA	62	Exchange Memory and A	20
ARITHMETIC			
ADD	55	Add	20
ADC	57	Add with Carry	20
ADM	63	Add to Memory	20
MIN	61	Memory Increment	21
SUB	54	Subtract	21
SUC	56	Subtract with Carry	21
MUL	64	Multiply	21
DIV	65	Divide	22
LOGICAL			
ETR	14	Extract (AND)	22
MRG	16	Merge (OR)	22
EOR	17	Exclusive OR	23
REGISTER CHANGE			
CLA	0 46 00001	Clear A	23
CLB	0 46 00002	Clear B	23
CLAB	0 46 00003	Clear AB	23
CLX	2 46 00000	Clear Index	23
CLEAR	2 46 00003	Clear A, B, and Index	23
CAB	0 46 00004	Copy A into B	23
CBA	0 46 00010	Copy B into A	24
XAB	0 46 00014	Exchange A and B	24
ABC	0 46 00005	Copy A into B, Clear A	24
BAC	0 46 00012	Copy B into A, Clear B	24
CAX	0 46 00400	Copy A into Index	24
CXA	0 46 00200	Copy Index into A	24
XXA	0 46 00600	Exchange Index and A	24
CBX	0 46 00020	Copy B into Index	24
CXB	0 46 00040	Copy Index into B	24
XXB	0 46 00060	Exchange Index and B	24
STE	0 46 00122	Store Exponent	24
LDE	0 46 00140	Load Exponent	25
XEE	0 46 00160	Exchange Exponents	25
CNA	0 46 01000	Copy Negative into A	25
BRANCH			
BRU	01	Branch Unconditionally	25
BRX	41	Increment Index and Branch	25
BRM	43	Mark Place and Branch	25
BRR	51	Return Branch	26
BRI	11	Branch and Return from Interrupt Routine	26
TEST/SKIP			
SKE	50	Skip if A Equals Memory	27
SKG	73	Skip if A Greater than Memory	27
SKM	70	Skip if A Equals Memory on B Mask	27
SKA	72	Skip if A and Memory do not Compare Ones	27

Mnemonic	Code	Name	Page
TEST/SKIP (cont.)			
SKB	52	Skip if B and Memory do not Compare Ones	28
SKN	53	Skip if Memory Negative	28
SKR	60	Reduce Memory, Skip if Negative	28
SKD	74	Difference Exponents and Skip	28
SHIFT			
RSH	0 66 00xxx	Right Shift AB	29
LRSH	0 66 24xxx	Logical Right Shift AB	29
RCY	0 66 20xxx	Right Cycle AB	29
LSH	0 67 00xxx	Left Shift AB	29
LCY	0 67 20xxx	Left Cycle AB	29
NOD	0 67 10xxx	Normalize and Decrement X	29
CONTROL			
HLT	00	Halt	30
NOP	20	No Operation	30
EXU	23	Execute	30
BREAKPOINT TEST			
BPT1	0 40 20400	Test Breakpoint 1	30
BPT2	0 40 20200	Test Breakpoint 2	30
BPT3	0 40 20100	Test Breakpoint 3	30
BPT4	0 40 20040	Test Breakpoint 4	30
OVERFLOW			
OVT	0 22 00101	Overflow Indicator Test and Reset	31
OTO	0 22 00100	Overflow Indicator Test Only	31
REO	0 22 00010	Record Exponent Overflow	31
ROV	0 22 00001	Reset Overflow Indicator	31
INTERRUPT			
EIR	0 02 20002	Enable Interrupts	13
DIR	0 02 20004	Disable Interrupts	13
IET	0 40 20004	Interrupt Enabled Test	13
IDT	0 40 20002	Interrupt Disabled Test	13
AIR	0 02 20020	Arm/Disarm Interrupts	13
CHANNEL CONTROL			
ALCW	0 02 50000	Alert Channel W	37
DISW	0 02 00000	Disconnect Channel W	38
ASCW	0 02 12000	Alert to Store Address in Channel W	38
TOPW	0 02 14000	Terminate Output on Channel W	38
CHANNEL TESTS			
CATW	0 40 14000	Channel W Active Test	44
CETW	0 40 11000	Channel W Error Test	44
CZTW	0 40 12000	Channel W Zero Count Test	44
CITW	0 40 10400	Channel W Interrecord Test	44
INPUT/OUTPUT			
EOM	02	Energize Output M	36
EOD	06	Energize Output D	36
SKS	40	Skip if Signal Not Set	36
PIN	33	Parallel Input	37
POT	13	Parallel Output	37
MIW	12	Memory into W Buffer	50
WIM	32	W Buffer into Memory	50

Price: $3.50

XDS 940 COMPUTER REFERENCE MANUAL

90 06 40C

October 1969

XDS

Xerox Data Systems/701 South Aviation Boulevard/El Segundo, California 90245

Printed in U.S.A.

REVISION

This publication, 90 06 40C, is a revision of the XDS 940 Computer Reference Manual, 90 06 40B. Changes to the previous manual are indicated by a vertical line in the margin of the affected page.

RELATED PUBLICATIONS

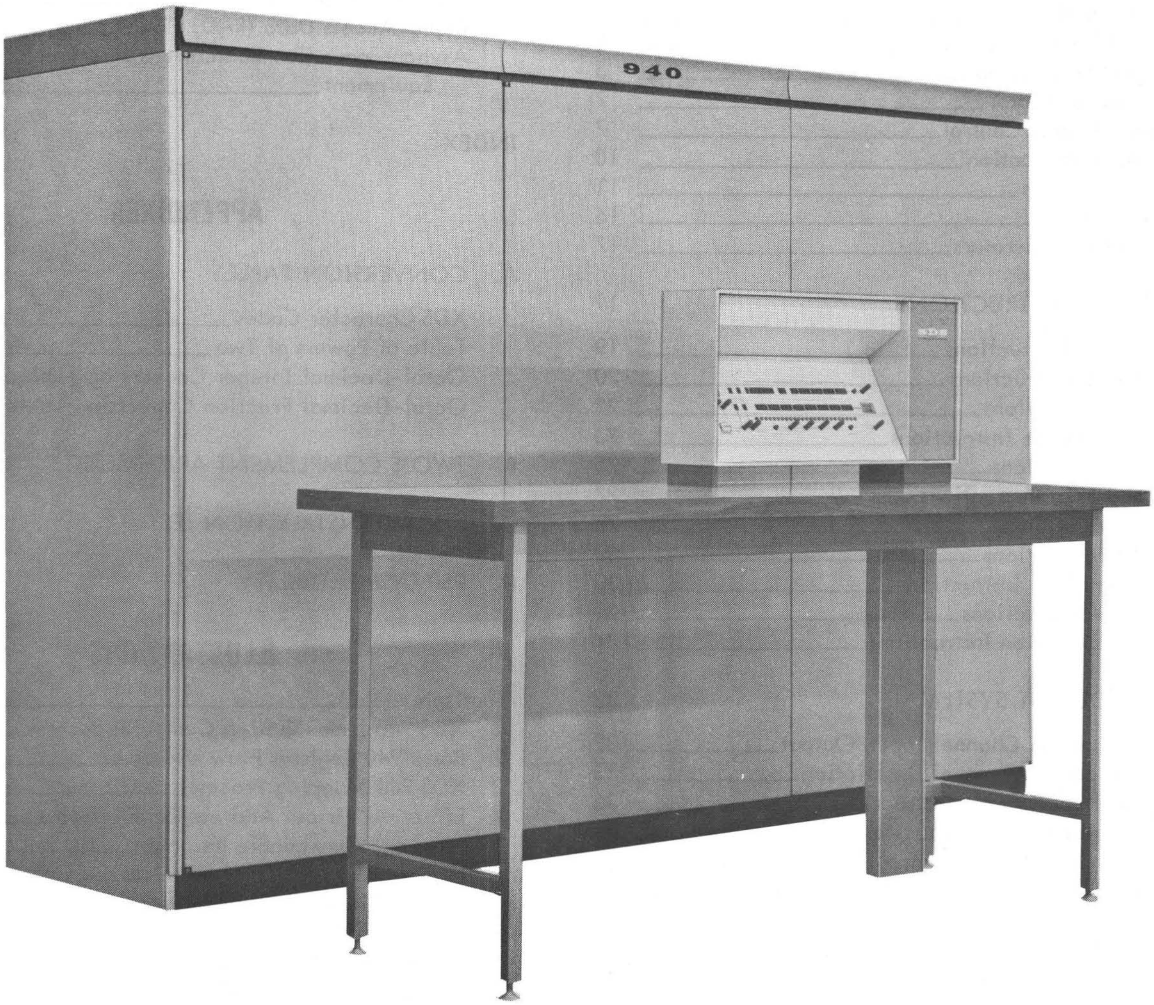
Title	Publication No.
XDS 920/930 Programmed Operators Technical Manual	90 00 20
XDS 940 Computer Diagnostic System Technical Manual	90 03 34
XDS 900 Series FORTRAN II Operations Manual	90 05 87
XDS 940 FORTRAN II Reference Manual	90 11 10
XDS 940 BASIC Reference Manual	90 11 11
XDS 940 QED Reference Manual	90 11 12
XDS 940 DDT Reference Manual	90 11 13
XDS 940 CAL Reference Manual	90 11 14
XDS 940 FORTRAN IV Reference Manual	90 11 15
XDS 940 Time-Sharing System Technical Manual	90 11 16
XDS 940 TAP Reference Manual	90 11 17
XDS 940 Terminal User's Guide	90 11 18
XDS 940 FORTRAN II Technical Notes	90 11 42

CONTENTS

APPENDIXES

ILLUSTRATIONS

TABLES



XDS 940 Time-Sharing Computer

1. XDS 940 SYSTEM

XDS 940 CHARACTERISTICS

The XDS 940 is a high-speed, general-purpose digital computer that is especially designed for large scale, time shared computing applications. The computer is an extension of the XDS 930 and, as such, it is fully compatible with all other XDS 900 Series Computers (see Appendix D).

- The XDS 940 has special features that enable it to meet the demanding requirements of a time-sharing computing environment:
  - o Monitor and user modes of operation provide for the establishment of and control over a set of privileged instructions that are reserved to the monitor mode. The privileged instruction set precludes user execution of any instruction that affects peripheral equipment, halts the computer, or changes the mode to the monitor state without relinquishing control of the computer to a monitor-mode program.
  - o A hardware-implemented "memory map" lets a monitor-mode program dynamically allocate memory and dynamically relocate and operate user-mode programs within scattered fragments of memory. This feature permits programs to reference procedures and data independently of their location in physical memory. It also provides the memory-protection features required by a time-sharing environment.
  - o System Programmed Operators (SYSPOPs) are single-word, instruction-format, direct entrances to various service routines normally provided with the operating system. SYSPOPs provide direct access to these services without requiring intervention of the operating system. Their availability along with the memory map, makes it possible to include and efficiently call public routines as common procedures available to all programs.
  - o The provision of arbitrary interruptibility assures that no program can "hang up" the computer through the improper execution of an infinite indirect address chain or infinite EXECUTE instruction sequence.
- The XDS 940 includes the following features that enable the computer to perform rapid, scientific computation and sophisticated, real-time control:
  - o Memory is nonvolatile in event of power failure (optional power failure feature permits saving contents of programmable registers); each memory module is functionally independent and directly addressable, with an access time of 0.7 microseconds and a cycle time of 1.75 microseconds.
  - o Memory overlap between central processor and I/O with two memory modules; address interleaving between two or more memory modules increases the probability of memory overlap
  - o 24-bit word plus parity bit
  - o Parity checking of all memory and input/output operations
  - o Priority Interrupt System
  - o 930 operation mode, which makes the XDS 940 operate exactly as an XDS930, providing program interchangeability with other XDS 900 Series Computers
  - o Multiprecision programming facility
  - o Instructions are single address, with

    Index register
    Indirect addressing
    Programmed operators
- The versatile XDS 940 input/output system includes the following features:
  - o One to four I/O communication channels (with interlacing capability), time-multiplexed with computer operation, provide input/output rates of over 288,000 words per second. Time-multiplexed input/output channels operate upon either words or characters. A 6-bit character is the standard character size; 6- and 12-bit characters, or 6-, 12-, and 24-bit characters can be specified as desired.
  - o A direct memory access system allows input/output transfer to occur simultaneously with computer memory access, providing input/output rates of over 570,000 words per second.
  - o One to four direct access communication channels operate upon words and characters. These channels accept 6-, 12-, and 24-bit characters. The number of characters per word is specified by the external peripheral device.
  - o A data multiplex channel uses the direct memory access connection and accepts/transmits information from external devices, or subchannels, which may operate simultaneously; thus, externally controlled and sequenced equipment may perform input/output buffering and control operations rather than the computer.
  - o Input/output with scatter-write and gather-read.
  - o Searching of magnetic tapes and discs can be accomplished independently of other computer operations.

- A parallel word input/output system, in addition to the channels, facilitates operating asynchronously on certain types of information under program control.
- Up to 32,000 output control and input test signals.

- The comprehensive set of field-proven peripheral equipment available for use in a 940 system includes:
  - Keyboard printers, available with electromechanical paper tape reader/punch
  - Automatic typewriters
  - Rapid-access data files
  - Photoelectric paper-tape reader and paper-tape punch with spooler, mounted on cart
  - Magnetic tape units (IBM-compatible; binary and BCD), punched-card equipment, line printers, graph plotters
  - Communications equipment, teletype consoles

- The XDS 940 software system is an integrated set of programming elements that exploit the latest concepts of interactive multiprogramming. A generalized system, it permits user operation in languages ranging from a machine-oriented assembly language to a FORTRAN compiler. The operating system is geared to maximize both responsive service to the user and operating efficiency. In particular, maximum use is made of reentrant processes and common routines. The following software elements constitute the basic XDS 940 operating system:
  - The Time-Sharing Monitor prevents users from destroying or gaining unauthorized access to programs or data of other users. Typical of the functions included in the monitor are I/O services for user programs, scheduling of user program operations, program error processing, and program-to-program communication.
  - The Time-Sharing Executive permits the user to call for various on-line services that best suit the user's problem requirements and ability to operate the machine. The executive provides complete bookkeeping facilities for file storage and retrieval from secondary memory, usage accounting, and file security.

    Under the Time-Sharing Executive, XDS offers conversational FORTRAN IV, a comprehensive algebraic language that includes ASA FORTRAN IV plus many XDS extensions, and BASIC, a compiler developed for time-sharing computer users. Use of BASIC requires no previous knowledge of computers; it closely resembles standard mathematical notation and contains editing features, simple input/output procedures, and language diagnostics.

    These incremental compilers permit flexible debugging of FORTRAN and BASIC programs from remote terminals.

- In addition to the XDS-offered software, there is a comprehensive array of additional subsystems that can be provided through the XDS Users' Group Library. This library contains extensive interactive programming subsystems that have helped set the state of the art in time-sharing. The following are some of these subsystems:
  - The String Processing System is a package of subroutines that performs string reading, writing, lookup, and comparison.
  - System Programmed Operators (SYSPOPs) enable users to obtain "public" system services in a direct and efficient manner, without monitor intervention. They place an additional set of "machine instructions" at the user's disposal, without increasing the user's memory allocation.
  - CAL (Conversational Algebraic Language) provides rapid solutions for small numerical problems in a highly interactive environment. It relieves the user of all concern with storage allocation for both programs and data, and offers a problem-oriented language for conversational use.
  - QED is a generalized text editor that allows the on-line user to create and modify symbolic text for any purpose. QED includes inserting, deleting, and changing lines of text; a line-edit feature; a powerful symbolic search feature; automatic tabs, which the user can set; and ten string buffers. The user can automatically save a set of editing commands for repeat execution later (clichés).
  - FORTRAN II has all the features of the standard XDS 900 Series FORTRAN II and can accept symbolic source-language input created on-line by QED; thus, an on-line compile-execute-edit-recompile cycle is easily achieved in the system.
  - TAP (Time-Sharing Assembly Program) is a 2-pass assembler with facilities for subprograms, literals, and macro instructions. It resembles the standard XDS Meta-Symbol assembler. The output, which can be directly processed by the debugging program, DDT, provides symbol tables for effective program checkout in terms of the source language symbols.
  - DDT is a versatile, sophisticated on-line debugging package that permits the user to examine, search, change, insert breakpoint instructions and step-trace his program at the symbolic level. DDT permits the use of literals in the same manner as the assembler; it can load both absolute and relocatable assembler-produced files, and its command language is geared to rapid interactive operation by the on-line user.

940 SYSTEM CONFIGURATION

The XDS 940 system, Figure 1, can be supplied in a variety of configurations to meet the requirements of diverse applications.

- The minimum hardware configuration required by the XDS 940 Time-Sharing software system contains the following equipment:

  - XDS 940 Computer, including a built-in time-multiplexed communications channel (TMCC) with two levels of priority interrupt
  - Memory interlace control unit for built-in TMCC
  - Two 16,384-word core memory modules
  - Multiple access to memory unit for each memory module
  - Direct-access communications channel (DACC), with two levels of priority interrupt
  - Two rapid-access disc (RAD) storage units (and couplers), with 4,194,284 characters of storage. This RAD storage is minimal; two additional RAD storage units can be attached to a single controller, making a total of 8,388,568 characters of storage.
  - Disc file (and coupler), with 67 million characters of storage
  - Magnetic tape control unit for 1 to 8 tape transports
  - Two magnetic tape transports: 75 ips, 200 characters per inch, 15 kHz (higher-performance units may be used)
  - Real-time clock, with two levels of priority interrupt
  - Interrupt control system, with arming feature and five levels of priority interrupt
  - One or more asynchronous communications controllers, each providing for up to 64 full-duplex, asynchronous teletype lines, one line per user station
  - Two (or more) keyboard/printers arranged for split operation for each user station. (Printer should be independent of keyboard and operated on a full-duplex circuit. Proper provision must be made for tying into the computer's communication system, either locally, or remotely through private lines or the switched network.)

- Besides the equipment required for the time-sharing software system, an XDS 940 installation requires some form of paper tape or card input for diagnostic purposes. If the XDS 940 user intends to operate the 940 as a 930, using standard XDS software, the system must be supplied with paper tape or card input/output capability at a performance level suitable for its efficient use in this mode. Under these conditions, the appropriate one of the following additional equipment sets is strongly recommended.

  - For diagnostic support:

    400-card/minute card reader

    300-card/minute card punch

- The XDS 940 time-sharing computer system permits expansion in several areas to handle increasing system requirements.

  - Additional memory modules, bringing the total amount of core storage up to a maximum of 65,536 words, can be provided.
  - As previously noted, system performance is greatly improved through use of a second RAD storage unit, and up to four such units can be readily accommodated. Expansions beyond this capacity require additional RAD controllers.
  - The peripheral processing capability of the central computer installation can be expanded by additional standard XDS devices, including line printers, card punches, paper tape units, and display equipment. The power fail-safe option can be added to further assure system integrity.

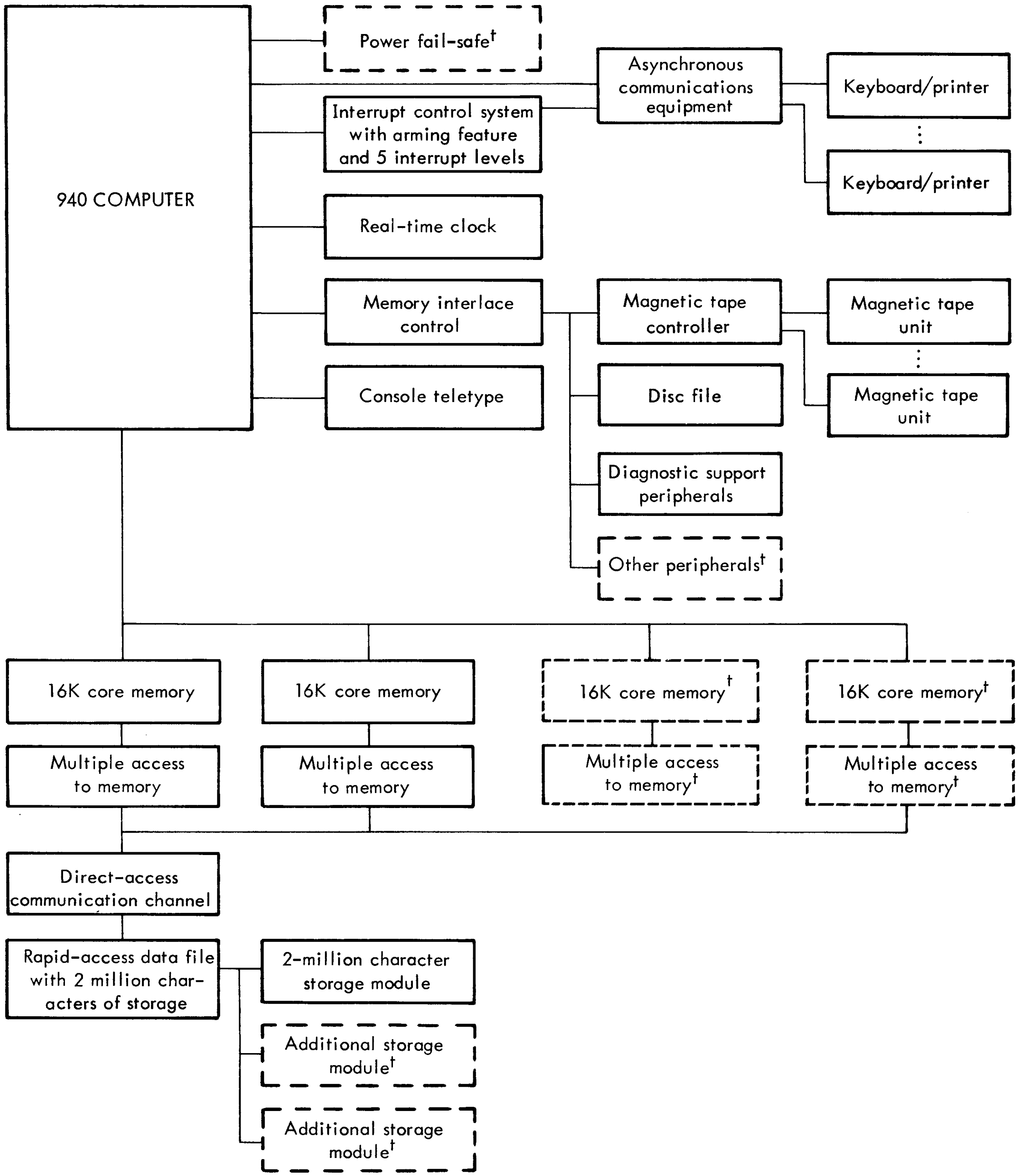


Figure 1. XDS 940 Time-Sharing Computer System

†Optional unit.

2. 940 CENTRAL PROCESSING UNIT

OPERATING MODES

The XDS 940 operates in three modes, which are designated as the normal mode, the user mode, and the monitor mode. In the normal mode, the 940 is completely compatible with the XDS 930 standard software systems (see Appendix D).

The primary feature that distinguishes the user and monitor modes from the normal mode is the automatic use of the memory map. The memory is considered to be divided into 32 blocks, each block containing 2048 words. In the user and monitor modes, memory is accessed under the control of the memory map. The memory map converts virtual addresses (i.e., addresses within the virtual machine in which the program assumes that it is operating) to actual addresses (i. e., actual physical core memory locations occupied by the program and its data). The memory map thus permits memory fragmentation by allowing the program to be located in noncontiguous blocks, which appear contiguous to the program. The address field of the 940 contains 14 bits; thus, 16,384 words (eight blocks) of memory are directly addressable by any program. Several techniques are available in the executive system to allow programs to address more than 16,384 words of memory.

The principal differences between the normal mode and the user mode are:

1. A set of privileged instructions is defined and forbidden in the user mode. This set consists of all undefined order codes, halt, all input/output orders, and all sense orders.
2. A new class of operations called System Programmed Operators (SYSPOPs) is provided. SYSPOPs are an extended form of the standard SDS Programmed Operators (POPs). The user can still use POPs unique to his application. If he does so, he must reserve space in his portion of memory for the POP transfer vector and for the routines that these POPs invoke. The SYSPOPs, however, permit user access to public routines provided by the operating system. As such, they do not occupy any space in the user's memory area. This, in effect, greatly augments the power of the machine that the user has at his disposal without preempting any of his allocated memory space.
3. All memory accesses made in the user mode go through the user memory map.

In the monitor mode, the full complement of 940 instructions is at the program's disposal. Two features distinguish this mode from the normal mode:

1. Monitor-mode programs can address memory through the user's memory map, thus giving them access to information in user areas utilizing user addresses. To accomplish this, a specific bit in the instruction word (or in any intermediate indirect address word) invokes the use of the user map for the duration of the instruction. In addition, the monitor mode is provided with a partial memory map, which maps only the high-order 4K addresses of a monitor-mode program. If a monitor-mode instruction does not invoke the user map, the monitor map is automatically invoked for monitor-mode addresses in the range 12K to 16K-1.
2. There is a minor difference in the storage position of the overflow indicator at the time of performing subroutine entries; this applies only to the monitor mode.

XDS 940 REGISTERS

The 940 registers of primary importance to the programmer are shown in Figure 2.

The A register (24 bits) is the main accumulator of the computer. The B register (24 bits) is an extension of the A register. The B register contains the less significant portion of double-length numbers.

The X register (24 bits) is used in address modification. Indexing operations occur only with the 14 least significant bits of the X register.

The C register (24 bits) is an arithmetic and control register. All instructions come from memory to the C register for decoding. Address modification and parity generation/detection take place in the C register.

The P register (14 bits) contains the virtual address of the current instruction. Unless modified by the program, the contents of P increase by one at the completion of each instruction.

Registers EM3 and EM2 are 3-bit registers that specify the portion of extended memory being used when the computer is operating in the normal or the monitor mode (see "Memory Extension System").

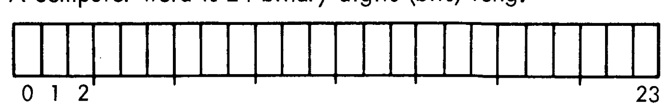
The user map is a 48-bit register that provides for dynamic relocation of user programs, for fragmentation of memory, and for two modes of memory protection (see "User Memory Map").

The monitor map is a 10-bit register that provides for dynamic relocation of up to 4096 words of the Monitor program (see "Monitor Memory Map").

The S register (16 bits) contains the actual address of the memory location to be accessed for instructions or data.

MEMORY WORD FORMAT

A computer word is 24 binary digits (bits) long.



The format above numbers the bits from the most significant to the least significant end of the word. Since one octal digit represents three binary digits, octal notation most easily represents the 24 bits of a word. In this manual, octal numbers are identified by the letter "B" following the least significant octal digit of the number.

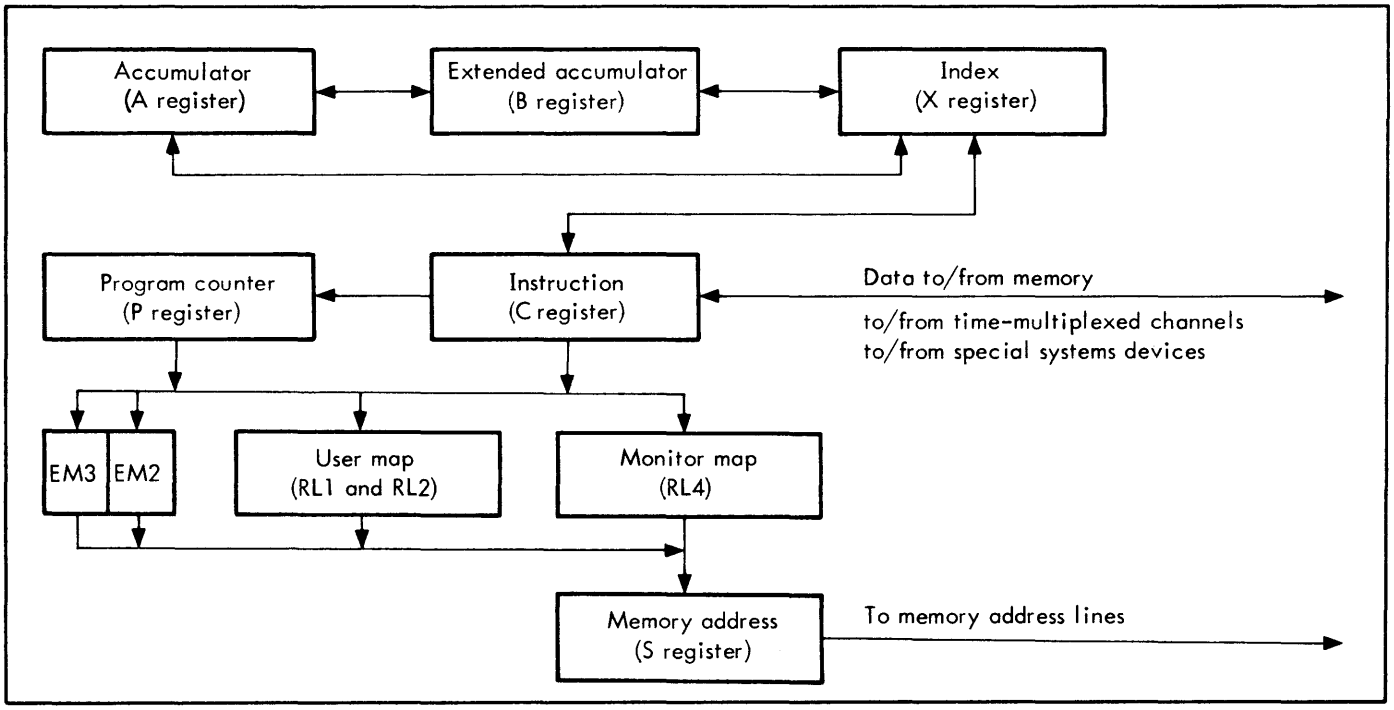
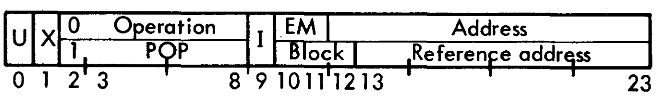


Figure 2. Basic 940 Register Flow Diagram

INSTRUCTION FORMAT

The computer instruction word format is:



The functions of the various portions of the instruction word are as follows:

Bit(s)	Function
0	When the computer is operating in the normal (930 compatible) mode, bit position 0 contains the relative address bit. The central processor decoding logic does not use or sense this bit in the normal mode. A 1 in this bit position causes some standard 930 loading programs to add the assigned location of the instruction to the addresss field contents prior to actual storage into the assigned location. A 0 in this bit position causes the loading program to store the instruction in the assigned location without changing the address field.
	When the computer is operating in the user mode, bit position 0 is ignored unless bit position 2 contains a 1; in which case, a 1 in bit position 0 of the instruction invokes the System Programmed Operator (SYSPOP) feature (see "System Programmed Operators").
	When the computer is operating in the monitor mode, bit position 0 determines whether the monitor map or the user map is to be used in forming the actual memory address of the instruction (see "Monitor Memory Map").
1	Bit position 1 is used to invoke indexing in all operating modes (see "Indexing").
2-8	If bit position 2 contains a 0, bit positions 3 through 8 contain the operation code of the machine instruction to be performed (this applies to all operating modes).
	When the computer is operating in the normal or monitor mode, a 1 in bit position 2 invokes the Programmed Operator (POP) feature (see "Programmed Operators").
9	Bit position 9 is used to invoke indirect addressing of memory-referencing instructions in all operating modes (see "Indirect Addressing").
10-23	Bit positions 10-23 contain an address value in the range 0 through 16K-1 (where K = 1024). When the memory system contains more than 16K words of storage, the memory extension registers provide for address values in the range 0 through 32K-1 (see "Memory Extension System"). When the address value in bit positions 10-23 is 8K or greater, bit positions 10 and 11 select the appropriate memory extension register (EM3 or EM2).
	In the user mode, bit positions 10-12 constitute a virtual memory block number and bits 13-23 specify a location within the virtual memory block.

Bits(s)	Function
	The 3-bit virtual block number is replaced by a 5-bit actual block number from the user memory map whenever a memory access is required by the instruction (see "User Memory Map"). In the monitor mode, bit position 0 of the instruction word (or any indirect access) determines whether the user memory map or the monitor memory map is to be used to obtain the actual address required for the instruction (see "Mode-Changing Capability"). If the user map is invoked by bit position 0, the 3-bit virtual block number is replaced by a 5-bit actual block number from the user memory map. If the user map is not invoked, address values in the range 0 through 12K-1 are treated as actual addresses (i.e., no replacement is made for block numbers 0 through 5), and address values in the range 12K through 16K-1 are treated as virtual addresses (i.e., virtual block numbers 6 and 7 are replaced by 5-bit actual block numbers from the monitor memory map).

FIXED-POINT DATA FORMAT

Fixed-point data words have the format:

±	Two's complement fraction
0	1 ... 23

Bit position 0 is the sign bit, with negative numbers having a 1 in bit position 0 and positive numbers having a 0 in bit position 0. The memory holds fixed-point numbers as 23-bit fractions with an assumed binary point to the left of bit position 1; the computer operates on these numbers arithmetically in a two's complement number system (see Appendix B for a discussion of two's complement arithmetic). Numbers held in one word have the equivalent precision of over six decimal digits. The range of values of the fixed-point format is from minus one to strictly less than plus one. Scaling is used in handling numbers during computation. Programmers sometimes consider fixed-point numbers to be integers with the binary point to the right of bit position 23. The range of integer values is from -2^{23} (-8,388,608) to $2^{23}-1$ (8,388,607).

OVERFLOW INDICATOR

The overflow indicator in the computer permits the detection of erroneous arithmetic operations that occur during the execution of a program. The overflow indicator is set to 1 (turned on) if any of the following occurs:

1. a sum or difference resulting from an addition or subtraction cannot be contained within the A register
2. multiplication of 40000000B by 40000000B (the A and B registers cannot contain this product)
3. a division with the absolute value of the numerator equal to or greater than the absolute value of the denominator (the A register cannot contain this quotient)
4. an arithmetic left-shift operation that changes the value of the bit in the sign position of the A register
5. bit 14 of the index register is not equal to bit 15 of the index register when the instruction RECORD EXPONENT OVERFLOW (ROV) is executed.

The 940 instruction set contains instructions to reset, or test and reset the state of the overflow indicator (see Section 3, "Overflow Instructions")

MEMORY ACCESS CONTROL

The control of a program's access to core memory is dependent upon the operating mode. In the normal mode, the memory extension system controls the accessing of memory systems that contain 16K or 32K words. In the user mode, the user memory map provides for relocation, fragmentation, and two modes of memory protection, for programs of up to 16K words of a memory system that may contain as many as 64K words. In the monitor mode, the monitor memory map provides for partial relocation and fragmentation of the executive program.

MEMORY EXTENSION SYSTEM

The memory extension system allows a normal-mode program to address memory locations 0 through 32K-1. The program always addresses the first 8K words of memory (locations 0 through 17777B) without regard to the memory extension system. The program invokes memory extension register EM2 whenever the 2 high-order address bits (bit positions 10 and 11 of the instruction) of a memory reference are 1 and 0, respectively (i.e., whenever a core memory location whose address is in the range 20000B through 27777B is referenced). Likewise, the program invokes memory extension register EM3 whenever the 2 high-order bits of a memory reference are both 1's (i.e., whenever a core memory location whose address is in the range 30000B through 37777B is referenced). The use of the memory extension registers applies to all memory references associated with an instruction execution, including the program counter (P register) and indirect addressing. However, the memory extension system does not apply to memory addresses used by a communication channel interlace operation (see Section 4, "Communication Channel Description").

The memory extension registers each contain a 3-bit value that becomes the 3 high-order bits of a 15-bit memory address when the appropriate memory extension register is invoked. Whenever the computer operator presses the START switch on the 940 control panel, the computer places the value 2 in register EM2 and places the value 3 in register EM3. This allows the program to directly address 16K words of memory (locations 0 through 37777B) without being conconcerned with the memory extension registers.

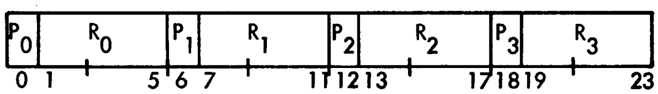
The memory extension registers can be set to any value in the range 0 through 7 by a specific configuration of the instruction ENERGIZE OUTPUT D (EOD), which simultaneously sets both EM2 and EM3 (see Section 3, "Memory Extension Instructions"). For example, if registers EM2 and EM3 contain the values 4 and 5, respectively, all instruction and program counter references in the range 20000B through 37777B are replaced by references in the range 40000B through 57777B. Thus, the instruction LDA 20300B would load the accumulator with the contents of location 40300B, and the instruction BRU 30504B would cause the next instruction to be taken from location 50504B.

When register EM3 does not contain the value 3, the computer lights the EM3 indicator on the control panel. When register EM2 does not contain the value 2, the computer lights the EM2 indicator on the control panel. These indicators can be tested by a specific configuration of the SKIP IF SIGNAL NOT SET (SKS) instruction (see Section 3, "Memory Extension Instructions"). The values in memory extension registers are stored in a "mark" location when a MARK PLACE AND BRANCH (BRM) instruction is executed (see Section 3, "Branch Instructions").

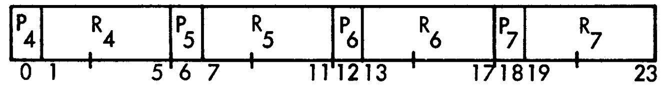
USER MEMORY MAP

The user memory map provides for dynamic relocation of programs, for fragmentation of memory, and for two modes of memory protection. It is used to convert virtual addresses (i.e., addresses within the virtual machine in which the user's program assumes that it is operating) to actual addresses (i.e., actual physical core memory locations occupied by the user's program and data). To accomplish this, the memory map operates on the 14-bit 940 address field, which permits user programs to directly address 16,384 words of core memory. The user memory map consists of eight 6-bit quantities held in two 24-bit active circuit registers designated as RL1 and RL2. The structure of these registers is as follows:

RL1



RL2



A virtual address is converted through the memory map to an actual address in the following manner. The value, i, defined by the three high-order bits of a 14-bit virtual address, is used to select the proper one of the eight quantities, R_i. The five bits of R_i then have appended to them the 11 low-order bits of the virtual address to form a 16-bit actual address. (This operation does not add any time to instruction execution.) Memory addresses obtained by mapping therefore permit up to 65,536 words of core memory in the system. The mapping process is illustrated by the example shown in Figure 3.

From this description it may be seen that the memory is considered to be divided into 32 blocks, each containing 2048 words. In the user mode, a memory is accessed under control of a 5-bit block number and an 11-bit address, which specifies a location within the 2048-word block. When mapping is invoked, the upper three bits of a virtual address constitute the virtual block number. The mapping hardware replaces the virtual block number, i, with an actual block number R_i, which may be different from time to time as the program is moved in and out of memory. Because of the relationship of the block number and reference address, the user program is not aware of the block structure of the memory. Thus, the mapping hardware permits memory fragmentation by allowing the user's storage to be located in noncontiguous blocks, which appear to the user and to the computer to be contiguous. Because the address field of the 940 contains 14 bits, 16,384 words (eight blocks) are directly addressable by any user at any one time. Several techniques are available in the executive system to allow users to use more than 16,384 words in their programs.

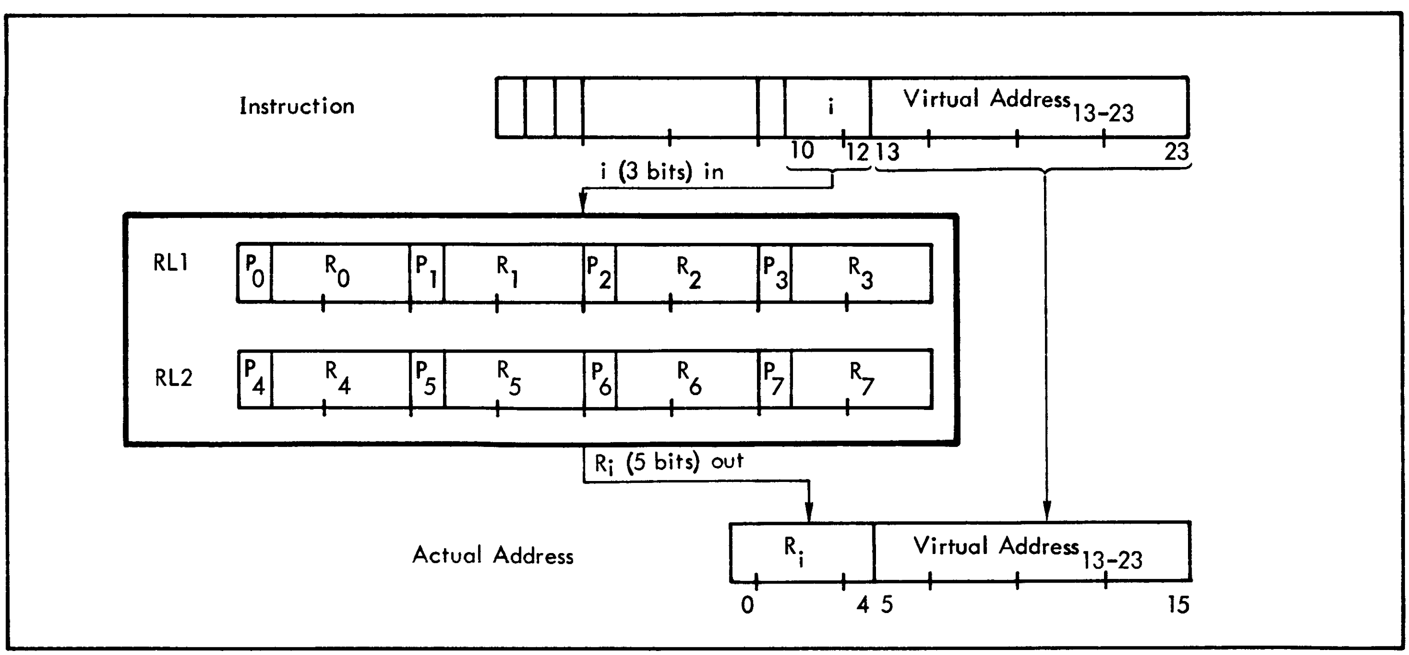


Figure 3. XDS 940 Mapping Process

The user memory map also provides two modes of memory protection. Only the five bits of the R_i quantity are used for actual block numbers. The sixth bit (the quantity P_i) designates a read-only block. The facility to have read-only storage enables users to share systems directly without interference and without the necessity of constantly calling the monitor to change the R_i quantities. Any Write request that involves a reference to a nonzero R_i quantity with a P_i bit of 1 results in a trap to location 43B (see "Trap System").

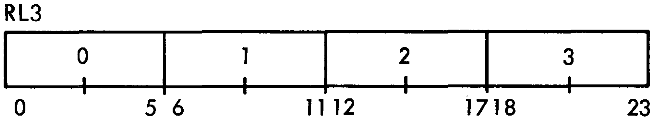
Absolute memory protection (i.e., protection against any reference) is accomplished by using $P_i = 1$ and $R_i = 0$ to mean that no memory is assigned to block i. Any reference to an R_i quantity with this value results in a trap to location 41B (see "Trap System").

The user memory map registers are loaded by an EOM-POT sequence. An EOM 20400B clears the RL1 register, and the following POT instruction loads it with a new 24-bit setting. Similarly, an EOM 21000B clears the RL2 register, and the following POT Instruction loads it with a new 24-bit setting. These operations require a total of eight memory cycles (14 microseconds).

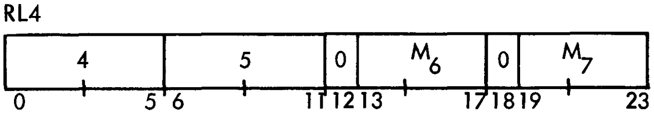
MONITOR MEMORY MAP

The 940 contains a partially implemented monitor memory map. For purposes of presentation, the monitor map may be considered to be laid out in a manner identical to that of the user memory map. That is, it may be thought of as consisting of eight mapping registers of six bits each, which are laid out in two 24-bit registers (designated as RL3 and RL4) as follows:

RL3



RL4

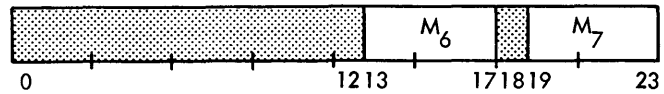


Actually, only the portions of the two registers labeled M_6 and M_7 are implemented with flip-flops. The other six registers may be thought of as though they permanently contained the following values:

(M_0)	=	00B	(M_3)	=	03B
(M_1)	=	01B	(M_4)	=	04B
(M_2)	=	02B	(M_5)	=	05B

M_6 and M_7 each actually contain five low-order flip flops. Thus, in the above figure, only a total of ten bits are implemented with flip-flops.

The monitor map registers M_6 and M_7 are loaded by an EOM-POT sequence. An EOM 21400B clears the M_6 and M_7 registers, and the following POT instruction loads them with new settings. The effective word of the POT instruction is assumed to be in the following format.



Bits 13-17 of the effective word are loaded into register M_6 and bits 19-23 of the effective word are loaded into register M_7. (Bits 0-12 and bit 18 of the effective word are ignored.)

In the monitor mode, if the sign bit of the instruction is a zero (so that mapping through the user map is not invoked), the address is mapped through the monitor map. This mapping is one-to-one for all addresses in the range 0 to 8K-1: monitor virtual addresses in this range always result in equivalent actual addresses.

For monitor virtual addresses in the range 8K to 12K-1, memory extension register EM2 is invoked. If EM2 contains the value 2, monitor virtual addresses in the range 8K to 12K-1 always result in equivalent actual addresses. However, if EM2 contains a value other than 2, the 2 high-order bits of the monitor virtual address (bit positions 10 and 11 of the instruction word) are replaced by the 3-bit contents of register EM2. Thus, register EM2 can be used to produce actual monitor addresses in the range 0 through 32K-1, just as if the computer were operating in the normal mode (see "Memory Extension System").

For monitor virtual addresses in the range 12K to 16K-1 memory extension register EM3 is not invoked (as it would be were the computer in the normal mode). Instead, for addresses in the range 12K to 14K-1 the monitor mapping is through the 5 bits contained in M_6 and for monitor virtual addresses in the range 14K to 16K-1 the monitor mapping is through the 5 bits contained in M_7. The only access limitation imposed by the monitor map is that physical locations 0 to 2K-1 cannot be accessed using M_6 or M_7. A zero value in M_6 or M_7 causes the same result as a 40B value in R_0 through R_7, namely an out-of-bounds trap when their respective pages are accessed through these maps.

In actual implementation, if the computer is in the user mode, the user map is invoked. If the computer is in the monitor mode, then the sign bit of the instruction is inspected to determine whether or not to go through the user map. If this bit is a zero (so that the user map is not invoked), the two high-order bit positions of the address are inspected to see whether they contain the values 1 and 0, or 1 and 1, respectively. If they do not, the address is used as it stands. If they contain 1 and 0, respectively, then a high-order zero, the contents of EM2, and the 12 remaining address bits are used to form the 16-bit address (in the range 0 through 32K-1) of the actual location in memory. If the three high-order bit positions contain 1, 1, and 0, respectively, the five bits of M_6 are concatenated with the 11 remaining low-order bits of the address to obtain the 16-bit address of the actual location in memory. If they contain 1, 1, and 1, respectively, the five bits of M_7 are used in a similar fashion. The convention used in the last two cases overrides any use of the memory extension register when operating in the monitor mode.

Mapping through the user memory map for individual instructions can also be invoked in the monitor mode. When accessing memory in the monitor mode to obtain the effective address of an instruction, any word encountered with bit 0 set to 1 causes the user mapping operation to apply immediately and for the remaining duration of the instruction. Thus, in the monitor mode, an instruction with bit 0 set to 1 causes its address field to be taken through the user map, while an instruction with a chain of indirect addresses invokes user mapping the first time a 1 in bit position 0 occurs in an indirect address. In the latter case, subsequent indirect references also use the user map until the instruction is completed.

MODE-CHANGING CAPABILITY

The normal (930) mode is invoked whenever the computer is in idle and the START button on the computer control console is depressed. Transition to the normal mode can be effected only in this manner. The transition from normal to the monitor mode is made by executing an EOM 22000B. The transition from monitor to user mode is made by executing any branch instruction in which user mapping is invoked. The user can cause a transition from user to monitor mode only by executing a SYSPOP, which returns control to the executive system. An interruption or a rrap that occurs when the computer is in the user mode also causes the computer to revert to the monitor mode. There is no means for transferring directly from the normal to the user mode.

To provide proper subroutine returns, the previous mode of the machine is stored in bit position 0 of the subroutine link of both interrupt and SYSPOP routines. Since bit position 0 is the bit that invokes user mapping, when the return instruction is executed, the computer automatically reverts to the mode under which it was operating at the time of the interrupt or the execution of the SYSPOP. If arguments are accessed indirectly through the link, mapping is or is not applied, depending on the mode storage bit. Hence, SYSPOP routines, which operate in the monitor mode, will correctly address memory through the link – independent of the mode of the calling program. Thus, interrupt routines are independent of the mode of the machine at the time of the interrupt, and system routines explicitly called by the various programs do not require software interpretation of the mode of the calling program, the location of the call, the location of the arguments, or the specific action requested.

It should be noted that interrupt routines take no more time and, in fact, are no different from similar routines in a non-time-sharing system. Furthermore, the overhead associated with calls to the system (SYSPOPs) is only four memory cycles.

ADDRESS MODIFICATION

Indexing and indirect addressing, used singly or in combination, are used to perform address modification. In both indexing and indirect addressing, the computer performs address modification after bringing the instruction from memory but before executing it. The instruction and/or indirect addressing form the "effective virtual address". If the instruction fetches an operand from memory or stores a result in memory, the effective virtual address is converted to an actual address, as described above.

INDEXING

The computer contains an index (X register) for address modification. The use of this register to modify the address in an instruction does not increase instruction execution time. If the content of bit position 1 (the index bit) in an instruction is 1, the computer adds the contents of bit positions 10 through 23 of the X register to the address field of the instruction. This addition does not keep any overflow or carry beyond the fourteenth address bit.

The instruction set provides instructions for modifying and testing the X register, and for transferring information between the X and B registers, the X and A registers, and the X register and memory.

INDIRECT ADDRESSING

The indirect address bit is in bit position 9 of the instruction. This bit determines whether the computer uses indirect addressing with the instruction being executed.

A 0 in the indirect address bit position causes the computer to use the address field (as modified by indexing, if indexing was invoked by a 1 in bit position 1) as the effective virtual address of the instruction.

A 1 in the indirect address bit position causes the computer to access the contents of the location pointed to by the actual address (determined as described above) as if it were an instruction without an instruction code; that is, the address logic reinitiates address decoding, using the word in the actual location (the memory location whose address is the actual address). This is an iterative process and provides multilevel indirect addressing and indexing. (Indirect addressing adds one cycle time to instruction execution time for each level of addressing.) The programmer can use indexing to modify indirect addressing at any level, as shown in Figure 4.

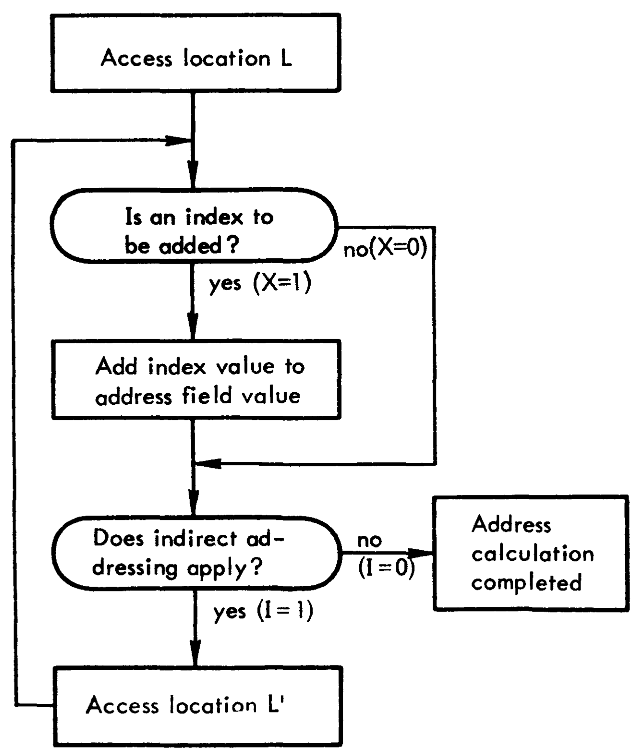


Figure 4. Effective Virtual Address Calculation

In this figure, location L contains an instruction whose address is the result of applicable operations involving use of the memory extension registers, the user memory map, or the monitor memory map. Similarly, L' represents an actual memory location whose address is the result of applicable operations involving use of the index register, the memory extension registers, the user memory map, or the monitor memory map.

HARDWARE HANG-UP PREVENTION

To continue to provide extremely rapid response to interrupts and to insure that user programs cannot inadvertently tie the computer up in an indefinitely long uninterruptible state, interrupt requests take precedence over indirect address calculations and EXECUTE instructions. If either operation is in progress when an interrupt request occurs, execution of the instruction is aborted and the interrupt request is acknowledged; the link to the interrupt routine contains the address of the aborted instruction. Upon return from the interrupt routine, the aborted instruction is restarted. This feature insures the system against indefinite hang-up caused by infinite address or EXECUTE loops in a user's program. This feature is operative only in the monitor and user modes.

INTERRUPT SYSTEM

The XDS 940 Computer contains a priority interrupt system that provides added program control of input/output and compute operations and allows immediate recognition of special external conditions on the basis of predetermined priority. The priority interrupt system is essentially a combination of hardware provisions and programming techniques. Various devices (such as the communication channels, power fail-safe, real-time clock) can cause interrupt of the program being executed by the computer by transmitting interrupt pulses to interrupt levels in the computer.

PRIORITY ASSIGNMENT

All interrupt devices used with a special computer installation are assigned unique, numbered priority levels (see Table 1) identified by octal numbers 30B through 37B, 56B through 75B, and 200B through 1777B, with the higher priority levels having a smaller number. The optional power fail-safe interrupt levels (36B and 37B) and the memory parity levels (56B and 57B) are "override" interrupt levels; they have the highest priority of all. Interrupt levels 30B-75B are optional hardware interrupt levels, normally added in pairs. Interrupt levels 200B-1777B are special systems interrupt levels that can be added in any number (up to 896) for general-purpose interrupts.

Table 1. Interrupt Location Assignments

Level	Name	Level	Name
30B	Channel Y Zero Word Count (End of Word)	70B	Channel G Zero Word Count
31B	Channel W Zero Word Count (End of Word)	71B	Channel G End of Record
32B	Channel Y End of Record (End of Transmission)	72B	Channel H Zero Word Count
33B	Channel W End of Record (End of Transmission)	73B	Channel H End of Record
⋮		74B	Clock Sync
36B	Power On	75B	Clock Pulse
37B	Power Off	⋮	
⋮		200B ⋮ 217B	Group 0 Optional Special Systems Interrupt (address code 00B)
56B	CPU Parity		
57B	Input/Output Parity		
60B	Channel C Zero Word Count (End of Word)	220B ⋮ 237B	Group 1 Optional Special Systems Interrupts (address code 01B)
61B	Channel C End of Record (End of Transmission)		
62B	Channel D Zero Word Count (End of Word)		
63B	Channel D End of Record (End of Transmission)	⋮	
64B	Channel E Zero Word Count	1760B ⋮ 1777B	Group 55 Optional Special Systems Interrupts (address code 67B)
65B	Channel E End of Record		
66B	Channel F Zero Word Count		
67B	Channel F End of Record		

INTERRUPT LEVEL OPERATION

As shown in Figure 5, each interrupt level has three distinct operating states. In the inactive state, the level has not received a pulse from its assigned interrupt device. When the pulse is received, the level is unconditionally set to the waiting state. If no higher-priority level is in the waiting or active states, the interrupt level causes the computer to execute the instruction stored in the memory location corresponding to the priority number of the interrupt level as the next instruction. This recognition of an interrupt signal by the computer is accomplished at the end of the execution cycle of the instruction currently being executed, and advances the interrupt level to the active state. The instruction in the interrupt location is executed without incrementing the program counter (P register), and all lower priority interrupt levels are inhibited until the interrupt level is cleared. The instruction placed in the interrupt location can be either a single instruction or a subroutine entry, depending on the type of interrupt level.

Single-Instruction Interrupt Level

If the interrupt level is a "single-instruction" interrupt level, the computer executes the instruction in the interrupt location, clears the interrupt level back to the inactive state, and executes the next instruction in sequence after the instruction at which the interrupt signal was acknowledged. For example, if a clock is connected to the computer so that it pulses an interrupt line at specified intervals, the program can maintain a real-time clock. If the clock is connected to interrupt level 75B (and location 75B contains the instruction MIN 2050B), the computer adds 1 to location 2050B each time the clock pulse causes an interrupt. The main program can examine location 2050B, whenever necessary, to determine how many time increments have elapsed, since the clock was started.

Thus one of the optional hardware interrupt levels and any of the interrupt levels 200B-1777B may be single-instruction interrupts, as required. Single-instruction interrupt levels require that the instruction have a timing of 2 or more cycles; otherwise, the interrupt level is not cleared after the single instruction is executed. Also, if the instruction in the interrupt location is a branch (and the branch should occur), the interrupt is cleared but there is no automatic return to the interrupted program.

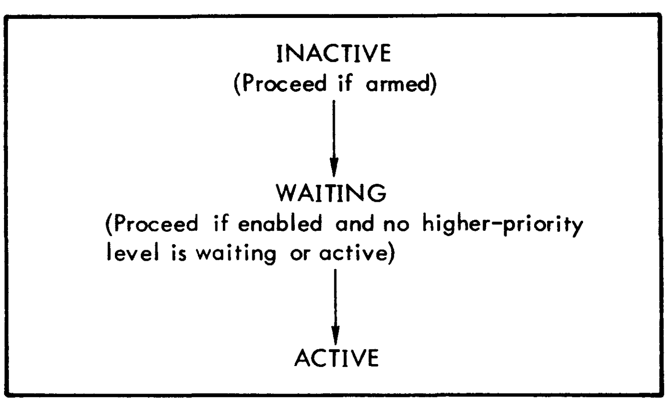


Figure 5. Interrupt Arm/Enable Response

Subroutine Interrupt Level

If the interrupt level is not a single-instruction level, it is a "subroutine" interrupt level; in which case, the instruction in the interrupt location is normally a MARK PLACE AND BRANCH (BRM) instruction to a servicing subroutine that ends in a BRANCH AND RETURN FROM INTERRUPT ROUTINE (BRI) instruction, addressed to the first location of the subroutine. The BRM instruction places the user map indicator, the overflow indicator, the current contents of the memory extension registers, and the current contents of the program counter (address of the next instruction in sequence after the interrupted instruction) in the first location of the servicing subroutine. During execution of the instructions within the servicing subroutine, a higher-priority interrupt can be acknowledged by the computer and the servicing subroutine is interrupted until the higher-priority interrupt has been processed. This allows interrupt levels to be arranged in the order of their importance and/or need for servicing. The BRI instruction at the end of the servicing subroutine clears the interrupt level back to the inactive state and returns program control to the next instruction in sequence in the interrupted program.

NONINTERRUPTABLE INSTRUCTIONS

If an ENERGIZE OUTPUT M (EOM) or ENERGIZE OUTPUT D (EOD) instruction is being executed, the computer does not acknowledge the interrupt signal until the instruction following the EOM or EOD is executed. Also, if an INCREASE INDEX AND BRANCH (BRX) instruction is being executed in the normal mode and the branch should occur, the computer does not acknowledge an interrupt signal until the instruction to which the BRX branches is executed (this restriction does not apply to BRX instructions being executed in the user or the monitor mode).

INTERRUPT ARM/ENABLE RESPONSE

Two control features concerning the interrupt system are available to the programmer. These features are arm/disarm and enable/disable. As shown in Figure 5, an interrupt level can proceed from the inactive to the waiting state only if it is "armed". If the level is "disarmed," the pulse is not "remembered" by the level. Once in the waiting state, the interrupt level remains in the waiting state until it has the highest priority of waiting interrupts. However, an interrupt level can proceed from the waiting to the active state only if the interrupt system is "enabled." If the interrupt system is disabled, the interrupt level remains in the waiting state until the interrupt system is enabled. Pressing the START button on the control console disarms, disables and clears all interrupt levels, and forces the computer into the normal mode.

Some computer applications require that certain conditions always be immediately recognized and acted upon by the computer. For this reason, certain interrupt levels are subject only to priority considerations, and always cause an interrupt if an interrupt device pulses its interrupt line. This type of an interrupt is considered always armed, always enabled, and cannot be disarmed or disabled, except by rewiring the computer. For example, the optional power fail-safe and memory parity interrupt levels are of this type. All communication channel interrupt levels are armed, disarmed, enabled, and disabled by means of the program.

Enable/Disable

The enable/disable feature is standard with the 940 computer, and operates on the entire interrupt system (with the exception of power fail-safe, memory parity, and real-time clock interrupt levels). The interrupt system is enabled by execution of ENABLE INTERRUPTS (EIR) and is disabled by execution of DISABLE INTERRUPTS (DIR).

EIR ENABLE INTERRUPTS (Privileged)

0	02	20002
0 2	3 8	9 23

This instruction is an EOM in the internal control mode (see Section 4, "Primary Input/Output Instructions") that turns on the INTERRUPT ENABLE indicator on the computer control panel and unconditionally enables the entire interrupt system. If any interrupt levels are in the waiting state when EIR is executed, the one with the highest priority proceeds to the active state.

DIR DISABLE INTERRUPTS (Privileged)

0	02	20004
0 2	3 8	9 23

This instruction turns off the INTERRUPT ENABLE indicator and unconditionally disables the entire interrupt system (with the exception of power fail-safe, memory parity, and real-time clock interrupt levels). If any interrupt levels are in the active state when DIR is executed, they are all processed in the order of their priority. If any interrupt levels are in the waiting state when DIR is executed, they will remain in the waiting state until EIR is executed.

IET INTERRUPT ENABLED TEST (Privileged)

0	40	20004
0 2	3 8	9 23

If the interrupt system is enabled when IET is executed, the computer skips the next instruction in sequence and executes the following instruction. If the interrupt system is disabled, the computer executes the next instruction in sequence (does not skip).

IDT INTERRUPT DISABLED TEST (Privileged)

0	40	20002
0 2	3 8	9 23

If the interrupt system is disabled when IDT is executed, the computer skips the next instruction in sequence and executes the following instruction. If the interrupt system is enabled, the computer executes the next instruction in sequence (does not skip).

Arm/Disarm

The arm/disarm feature is optional with the 940 computer, although some interrupt levels are always armed and some are individually armed by EOM instructions. If the optional Arm Interrupt Control Unit is present as a part of the computer, all interrupt levels from 200B to 1777B are armed and/or disarmed in groups of 16 interrupt levels (i.e., 200B-217B, 220B-237B, etc.), and only by a specific combination of the instructions ARM INTERRUPTS (AIR) and PARALLEL OUTPUT (POT). If the Arm Interrupt Control Unit is not present, these interrupt levels are considered to be always armed, but are still subject to control with enable/disable. Table 2 summarizes the arm/disarm feature for the various interrupt levels.

Table 2. Interrupt Arming Criteria

Level	Function	Arming Criteria
30B … 33B	TMCC Zero Word Count End of Record	Arm with EIR (compatible mode) or with input/output control mode EOM (extended mode)
36B, 37B 56B, 57B	Power Fail-Safe Memory Parity	Always armed
60B … 63B	TMCC Zero Word Count End of Record	(see 30B-33B above)
64B … 73B	DACC Zero Word Count End of Record	Arm with input/output control mode EOD
74B	Clock Sync	Always armed
75B	Clock Pulse	Arm with EOM 20100B, disarm with EOM 20200B
200B … 1777B	Special Systems	Arm with EIR or selectively arm/disarm with optional AIR

AIR ARM INTERRUPTS (Privileged)

0	02	20020
0 2	3 8	9 23

Air is an internal control EOM that prepares the Arm Interrupt Control Unit to receive a control word. The control word is transmitted to the control unit by a POT instruction (see Section 4, "Primary Input/Output Instructions"). The instruction sequence AIR-POT must be used for each group of 16 interrupt levels; otherwise, an unpredictable operation occurs. These instructions have no effect on the INTERRUPT ENABLE indicator, and the Control Unit is not affected by the indicator.

The control word which the instruction POT addresses has the following format:

Group code	C		Interrupt select bits
0 5	6	7	8 23

The group code (bits 0 through 5) identifies which group of interrupts is being addressed (e.g., an address code of 00B identifies interrupt levels 200B-217B). The C field (bits 6 and 7) specifies whether the interrupt levels selected by bits 8 through 23 of the control word are to be armed and/or disarmed:

6	7	Function
0	0	None
0	1	Arm only those interrupt levels that are selected by a 1 in bit positions 8-23. (Interrupt levels represented by a zero in bit positions 8-23 are not affected.)
1	0	Disarm only those interrupt levels that are selected by a zero in bit positions 8-23. (Interrupt levels represented by a 1 in bit positions 8-23 are not affected.)
1	1	Arm all interrupt levels selected by a 1 and disarm those levels selected by a zero in bit positions 8-23.

Bit position 8 of the control word represents the lowest numbered (highest priority) interrupt within the group identified by the address code (e.g., 200B, 220B, etc.). Bit position 23 represents the highest numbered (lowest priority) level within the group. See example below.

CHANNEL INTERRUPT DESIGNATIONS

As shown in Table 1, each I/O channel has two interrupt levels. These reflect the two distinct uses of interrupts during channel input and output. Also, each W, Y, C, and D channel level has two names that reflect their use in the extended or compatible I/O modes (see Section 4, "Compatible/Extended Input/Output Modes").

Single-Word Operations

A program can use channels W and Y as single-word, direct, program-controlled, input/output buffers. Special I/O instructions applicable to channels W and Y control this type of operation (see Section 4). In this mode, the program can specify that interrupt signals occur as each word is transferred from the channel buffer to the peripheral device on output, or as soon as the channel buffer is filled from the peripheral device on input; this is the end-of-word interrupt. The program can specify that an end-of-transmission interrupt occurs when the buffer detects a signal such as end-of-record from magnetic tape. During both input and output operations, this interrupt signal occurs when the peripheral device used in the transmission disconnects and the channel buffer becomes ready for another input/output operation.

Compatible Mode Interlaced Operations

The end-of-word and end-of-transmission interrupt levels also can control input/output termination for any time-multiplexed channel when the program is operating the channel buffers in the block transmission or "interlaced" compatible mode. In this mode, the end-of-transmission interrupt signal also occurs when the channel buffer has sent a specified number of words from memory to a peripheral device as well as when the channel detects an end-of-record signal. However, the end-of-word interrupt can occur on input only after a specified number of words have been read from a peripheral device, and then only if the channel buffer assembles another word. If the last specified word is read into memory before an end-of-record condition exists, the channel buffer reverts to the single-word mode of operation, in which case the end-of-transmission interrupt pulse occurs when the channel receives the end-of-record signal from the device. If either channel W or Y is being used, the remainder of the record can be stored in memory without the use of the interlace. However, channels C and D

Example:

The following partial program arms interrupt levels 210B-227B and disarms levels 230B-237B, but does not alter levels 200B-207B.

Location	Instruction	Address	Comments
	EIR		Enable entire interrupt system (turns INTERRUPT ENABLE indicator on)
	AIR		Prepare the Arm Interrupt Control Unit to receive a control word
	POT	CW1	Transmit the control word in location CW1 to the Arm Interrupt Control Unit
	AIR		An AIR must precede each POT
	POT	CW2	Transmit the control word in location CW2 to the Arm Interrupt Control Unit
	⋮		
CW1	DATA	200377B	This control word only arms levels 210B-217B. If any of levels 200B-207B are already armed or disarmed, they remain so.
CW2	DATA	1777400B	This control word arms levels 220B-227B and disarms levels 230B-237B, regardless of their previous state

operate with interlace only, and a data overrun (character rate error) may occur if the channel is neglected in this mode of transmission. If an end-of-record condition occurs first, only the end-of-transmission interrupt signal occurs. No end-of-word interrupt occurs during output. (See Section 4, "Compatible Mode Terminal Functions".)

The enable/disable instructions "enable and arm" or "disable and disarm" the end-of-word and end-of-transmission interrupt levels when the channel is operating in the compatible interlace mode. When the EIR instruction is executed, the interrupt system is enabled and these interrupt levels are also armed; when DIR is executed, the system is disabled and these interrupt levels are also disarmed.

Extended Mode Interlace Operations

When the XDS 940 input/output system uses channels within its full capabilities, special input/output functions control interlaced block transmission operations (see Section 4, "Extended Mode Terminal Functions"). The interrupt levels used with the extended input/output function control are zero word count and end of record. The zero-word-count interrupt signal occurs when the last of the number of words specified is placed into or brought from memory. The end-of-record interrupt signal occurs when the channel receives an end-of-record signal from the peripheral device. Input/output terminal functions can alter this latter occurrence for use with magnetic tapes. (See Section 4, "Extended Mode Terminal Functions".)

Effects of the Enable/Disable Feature on Armable Interrupts

When operating an input/output channel in the extended mode, the interrupt enable feature controls the armable interrupt levels (zero word count and end of record). If a channel generates an extended mode I/O interrupt signal while the interrupt system is disabled, the designated interrupt level goes to the waiting state. When the program again enables the interrupt system, the waiting interrupt level goes to the active state when its priority allows.

This feature greatly simplifies the programmer's handling of multiple-channel operations. The interrupt processing subroutine for one channel can disable the interrupt system while it processes the active interrupt level. During this time, the system receives all other interrupt signals in their respective levels, which go to the waiting state until the interrupt system is again enabled.

REAL-TIME CLOCK

The Real-Time Clock (RTC) provides a flexible time-orientation system for the XDS 940 Computer. It derives time pulses from the 60-Hz 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-designed source, thereby allowing time measurement to any required resolution for special applications. These timing marks are supplied at standard XDS 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 operates in either mode, depending only on the computer's stored program.

Location	Function	Type
74B	Clock sync	Normal
75B	Clock pulse	Single instruction

The clock sync interrupt level is always armed, but the clock pulse interrupt level can be armed and disarmed with the following privileged instructions.

Instruction	Action
EOM 20100B	Arm clock pulse interrupt level
EOM 20200B	Disarm clock pulse interrupt level

The clock pulse and clock sync interrupt levels function together to provide elapsed time, event counter, or time-of-day clocks.

Elapsed Time Clock

The elapsed time clock indicates 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 location contains the clock count (the two's complement of the number of time intervals to be counted), and the clock pulse interrupt location contains an SKR instruction whose effective address is the address of the location containing the clock counter.

Each clock pulse interrupt signal results in decrementing the absolute value of the clock count by one. When the clock count is zero, an interrupt signal is sent to the clock sync location. A supervisory or other appropriate control program can then be entered (via a BRM instruction) to perform the desired operation.

Continuously Incrementing Clock

The continuously incrementing clock maintains "time-of-day" for the computer. One memory location serves to count the timing marks. In this case, the clock pulse is used to increment this location. (The clock pulse interrupt location contains a MIN instruction.) When MIN is used as a single-instruction interrupt subroutine, it causes the contents of the effective location to be incremented by one. Furthermore, if the value of the new (incremented) contents of the effective location is zero, a clock sync interrupt signal is generated. A simple, straightforward subroutine (via a BRM instruction in the clock sync interrupt location) can be entered to reconstruct the exact time of day from the 24-bit count in the effective location of the MIN instruction.

AUTOMATIC POWER FAIL-SAFE SYSTEM

The 940 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, the power fail-safe 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. This solid-state system consists of ac power-sensing and memory sequencing circuitry, two high-priority interrupt levels (36B and 37B), and a "shut-down/start-up" programming sequence.

The SKIP IF SIGNAL NOT SET instruction (SKS 24000B) is an aid in programming this option. If the power-off interrupt (37B) has just occurred, the computer executes the next instruction in sequence (does not skip); otherwise, the computer skips the next instruction in sequence and executes the following instruction.

MEMORY PARITY INTERRUPTS

XDS 940 Computers incorporate an extensive memory parity checking system. The inclusion of parity generation and checking circuitry assures the integrity of data and instructions transferred among the memory, the central processing unit, and input/output channels.

In normal operation, the MEMORY PARITY 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 MEMORY PARITY indicator lighted; or the computer ignores the parity error and proceeds with the program (see Section 5).

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 enabled interrupts. One interrupt level (56B) is associated with the central processor; the other interrupt level (57B) is associated with the direct access communication channels and the data multiplexing system. Memory parity errors detected from these two sources produce an interrupt signal to the level associated with the cause. The processing routine associated with the interrupt level can then take appropriate action, such as reinitiate 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.

TRAP SYSTEM

When a condition that is to result in a program interrupt is sensed, a signal is sent to an interrupt level. If that level is armed, it advances to the waiting state. When the conditions for its acknowledgement have been achieved, the interrupt level advances to the active state, where it finally causes the computer to take an instruction from a specific location in memory. The computer may execute many instructions between the time that the interrupt-requesting condition is sensed and the time that the actual interrupt acknowledgement occurs. However, detection of any of the conditions listed in Table 3 results in a trap (the immediate execution of the instruction in a unique location in memory).

When a trap condition occurs, the instruction causing the trap condition is terminated with a trap sequence. In this sequence, the instruction in the location associated with the trap is executed. An interrupt acknowledgement cannot occur until the instruction in the trap location is executed. The instruction in the trap location must be a MARK PLACE AND BRANCH (BRM) instruction.

PRIVILEGED INSTRUCTION VIOLATION

The set of privileged instructions prohibited in the user mode is as follows:

Code	Mnemonic	Function
00B	HLT	Halt
02B	EOM	Energize output M
03B	none	undefined
04B	none	undefined
05B	none	undefined
06B	EOD	Energize output D
07B	none	undefined
10B	MIY	Memory into Y buffer
11B	BRI	Branch and return from interrupt routine
12B	MIW	Memory into W buffer
13B	POT	Parallel output
15B	none	undefined
21B	none	undefined
24B	none	undefined
25B	none	undefined
26B	none	undefined
27B	none	undefined
30B	YIM	Y buffer into memory
31B	none	undefined
32B	WIM	W buffer into memory
33B	PIN	Parallel input
34B	none	undefined
40B	SKS	Skip if signal not set
42B	none	undefined
44B	none	undefined
45B	none	undefined
47B	none	undefined

An attempt to execute a privileged instruction while the computer is in the user mode causes an immediate trap to location 40B. The address stored in the "mark" location (by the BRM instruction in location 40B) is the virtual address of the attempted privileged instruction.

UNAUTHORIZED MEMORY ACCESS

The user memory map provides protection against unauthorized memory accesses by user-mode programs (see "User Memory Map"). To prevent a user-mode program of fewer than 8 blocks from accessing a block outside its addressing range, those mapping registers associated with unused (and therefore prohibited) blocks are loaded with 40B. If any attempt (including indirect addressing) is made to reference a memory location having a virtual address pointing to an R_i with this content, a trap to location 41B will result.

Table 3. Summary of XDS 940 Trap System

Trap condition	Time of occurrence	Trap location	Program counter value stored by BRM
Privileged instruction violation	Instruction decode	40B	Virtual address of privileged instruction
Unauthorized memory access	Prior to memory access	41B	Virtual address of aborted instruction[†]
Memory protection violation	Prior to memory access	43B	Virtual address of aborted instruction[†]
Monitor-to-user transition	After transition	44B	Virtual address of next user instruction

[†]For trap conditions caused by a branch, the program counter value stored by the BRM instruction may be the virtual address of a prohibited or protected location (see "Unauthorized Memory Access" and "Memory Protection Violation").

If the instruction causing the unauthorized memory access is not a branch instruction, the address stored in the "mark" location (by the BRM instruction in location 41B) is the virtual address of the instruction attempting to access unauthorized memory.

If the instruction causing the unauthorized memory access is a branch instruction, the address stored in the "mark" location (by the BRM instruction in location 41B) is determined by the following table:

Instruction	Effective location authorized?	Program counter value stored by BRM
BRU	no	virtual address of BRU
BRX	no	virtual address of BRX
BRM	no	virtual address of BRM
	yes, but effective location + 1 is not	effective virtual address + 1
BRR	no	virtual address of BRR
	yes, but 1 + the address in the effective location is not	1 + address in effective location

The monitor memory map imposes an access restriction on monitor-mode programs that attempt to access actual locations in the range 0 through 2K-1 by using either of the monitor map registers M_6 or M_7 (i.e., a value of 0 in M_6 or M_7). In this case, the address stored in the "mark" location (by the BRM instruction in location 41B) is determined as shown above for user-mode programs. In addition, if the instruction BRI causes an unauthorized memory access condition, the address stored in the "mark" location (by the BRM instruction in location 41B) is determined as follows:

Effective location authorized?	Program counter value stored by BRM
no	virtual address of BRI
yes, but the address in the effective location is not	address in the effective location

MEMORY PROTECTION VIOLATION

The user memory map provides memory write protection with the high-order bit associated with each of the user map block numbers. Writing into any assigned actual block is allowed if and only if the high-order bit associated with the actual block number is 0. If the high-order bit is 1, the associated actual block is write protected, and any attempt to alter the contents of a location in the write-protected block results in a trap to location 43B. The program counter value stored in the "mark" location (by the BRM instruction in location 43B) is the virtual address of the instruction attempting to alter protected memory.

MONITOR-TO-USER TRANSITION

The monitor-to-user transition trap is effective only when it is enabled. The trap is enabled by execution of EOM 22400B by a monitor-mode program. If this trap is enabled, the computer traps to location 44B whenever the monitor-mode program performs a transition to the user mode (i.e., a branch instruction that invokes the user memory map). The program counter value stored by the BRM instruction in location 44B is the virtual address of the next instruction to be executed in the user-mode program. The monitor-to-user transition trap is disabled whenever any trap condition occurs; it remains disabled until it is again enabled by execution of EOM 22400B.

PROGRAMMED OPERATORS

The XDS Programmed Operator (POP) feature enables a programmer to code a subroutine call with a single instruction, just as if the subroutine were an actual machine instruction. When the computer detects a 1 in bit position 2 of an instruction, bit positions 2 through 8 of the instruction are not interpreted as an operation code; instead, they are treated as an address to which the computer transfers control; the address field of the instruction designates an address for use by the subroutine. There are 64 locations (100B through 177B) to which a transfer may occur. These 64 locations constitute a linkage table; they normally contain appropriate unconditional transfer (BRU) instructions to maintain the communication link between the POP code and the subroutine being called by it.

The location from which the transfer is made, at the time the computer detects the POP code (that is, the contents of the P register), is preserved in location 0 . Thus, the normal BRR instruction may be used to leave the POP subroutine and return to the main program. To allow access to the operand in the main program by the POP subroutine, bit position 9 (the indirect address bit) is unconditionally set to 1. In this manner, when the subroutine refers indirectly to location 0, the indirect addressing is perpetuated one more level, thereby enabling the subroutine to gain access to the operand in the main program.

A library of programmed operator subroutines is available to greatly extend the XDS 940 normal-mode instruction list. Each subroutine is specified by a unique mnemonic code and represents an available instruction that may be used directly in preparing normal-mode 940 programs. Up to 64 of these programmed operator instructions may be used to prepare any one normal-mode program.

The normal-mode program loading system automatically organizes the interconnection between the programmed operator instructions and the corresponding subroutines. Each programmed operator instruction mnemonic code is converted on input to an instruction code in the range 100B through 177B. A memory location from 100B through 177B corresponding to each assigned instruction code is loaded with an unconditional branch to the corresponding subroutine. Refer to the XDS 920/930 Computer Programmed Operators Technical Manual (XDS Publication 900020) for further details.

PROGRAMMED OPERATOR EXECUTION

Depending upon the operating mode (normal, user, or monitor), the following operations take place when the computer detects a programmed operator.

Normal Mode

1. Store the current value of the overflow indicator in bit position 0 of location 0.
2. Reset the overflow indicator to 0.
3. Store zeros in bit positions 1-8 of location 0.
4. Store a 1 in bit position 9 of location 0.
5. Store the current contents of the P register (address of the POP instruction) in bit positions 10-23 of location 0.
6. Load the POP code value into the P register.

User Mode

1. Store the current value of the overflow indicator in bit position 0 of virtual location 0.
2. Reset the overflow indicator to 0.
3. Store zeros in bit positions 1-8 of virtual location 0.
4. Store a 1 in bit position 9 of virtual location 0.
5. Store the current contents of the P register in bit positions 10-23 of virtual location 0.
6. Load the POP code value into the P register.

Monitor Mode

1. Bit 0 of the instruction word must be a 0.
2. Store the current value of the overflow indicator in bit position 2 of actual location 0.
3. Reset the overflow indicator to 0.
4. Store zeros in bit positions 0[†], 1, and 3-8 of actual location 0.
5. Store a 1 in bit position 9 of actual location 0.
6. Store the current contents of the P register in bit positions 10-23 of actual location 0.
7. Load the POP code value into the P register.

SYSPOP (SYSTEM PROGRAMMED OPERATOR)

The system executive program includes many complex services, some of which are of great potential value to a user. Such services are provided by SYSPOP calls. A SYSPOP is a user-mode POP instruction that contains a 1 in bit position 0. When a SYSPOP is encountered in the user mode, the 940 immediately reverts to the monitor mode before executing the operation. The user thus has the facility to jump to public service programs through the standard system transfer vector, which is outside his allocated memory space. The return link from a SYSPOP-entered routine automatically forces the system to the mode that existed upon execution of the SYSPOP; thus, SYSPOP routines can be used by programs in either the monitor or user mode with no loss of system control. In essence, the POP of the monitor mode is the SYSPOP of the user mode. Software traps in the privileged SYSPOP routines prevent user-mode programs from invoking restricted operations.

[†]Contains a 1 if entered via a SYSPOP.

3. MACHINE INSTRUCTIONS

This section contains a description of XDS 940 instructions, grouped by functional category. With the description of each instruction is a diagram representing the format of the instruction. Preceding this diagram is the assembler mnemonic code that identifies the instruction and the name of the instruction. If the instruction is not implemented in the 930 computer, the instruction is labeled "940 only". If the instruction is not executable while the computer is in the user mode, the instruction is labeled "privileged".

Within the instruction diagram, the following conventions are used.

1. The letter "U" in bit position 0 indicates that, in the monitor mode, the virtual memory address is mapped through the user map if this bit position contains a 1 and is mapped through the monitor map if this bit position contains a 0. This bit position is ignored in the normal mode and in the user mode.
2. The letter "X" in bit position 1 indicates that the instruction invokes indexing if bit position 1 contains a 1 (indexing adds no additional time to instruction execution). If the diagram contains a 0 in bit position 1, indexing does not apply to the instruction and an unpredictable operation occurs if indexing is attempted.
3. Bit positions 3-8 contain a 2-digit octal number that is the operation code of the instruction.
4. The letter "I" in bit position 9 indicates that the instruction invokes indirect addressing if bit position 9 contains a 1 (indirect addressing adds 1 memory cycle for each level). If the diagram contains a 0 in bit position 9, indirect addressing does not apply to the instruction and an unpredictable operation occurs if indirect addressing is attempted.

Following the description of the instruction is a symbolic list of all registers, indicators, and memory locations that can be affected by the instruction. The following symbols are used:

- A A register
- B B register
- AB Combined A and B registers
- X Index register
- P P (program counter) register
- Of Overflow indicator
- EL Effective location

Parentheses are used to denote "contents of". For example, "(A)" denotes "contents of the A register". The contents of registers and the addresses and contents of memory locations are expressed, in this manual, as octal numbers followed by the letter "B". All numbers (except in instruction diagrams) not followed by the letter "B" are decimal base.

Subscripted numbers identify inclusive bit positions. For example, "$(A)_{0-11}$" indicates "the contents of bit positions 0 through 11 of the A register".

All instruction times are given in memory cycles, each cycle being 1.75 microseconds, which includes accessing the instruction, indexing, and accessing the required operands.

LOAD/STORE INSTRUCTIONS

LDA LOAD A

U	X	0	76	I	Reference address
0	1	2	3–8	9	10–23

LDA loads the effective word into the A register.

Affected: (A) Timing: 2

STA STORE A

U	X	0	35	I	Reference address
0	1	2	3–8	9	10–23

STA stores the contents of the A register in the effective location.

Affected: (EL) Timing: 3

LDB LOAD B

U	X	0	75	I	Reference address
0	1	2	3–8	9	10–23

LDB loads the effective word into the B register.

Affected: (B) Timing: 2

STB STORE B

U	X	0	36	I	Reference address
0	1	2	3–8	9	10–23

STB stores the contents of the B register in the effective location.

Affected: (EL) Timing: 3

LDX LOAD INDEX

U	X	0	71	I	Reference address
0	1	2	3–8	9	10–23

LDX loads the effective word into the index register.

Affected: (X) Timing: 2

STX STORE INDEX

U	X	0	37	I	Reference address
0	1	2	3–8	9	10–23

STX stores the entire contents of the index register in the effective location.

Affected: (EL) Timing: 3

EAX COPY EFFECTIVE ADDRESS INTO INDEX

U	X	0	77	I	Reference address
0	1	2	3 8	9	10 23

In the normal and user modes, EAX copies the effective virtual address into bit positions 10-23 of the index register. The ten most significant bits of the index register (0-9) are unaffected in the normal and user modes.

The process of computing an effective address for this instruction operates as in a LOAD A instruction, except that instead of obtaining the contents of the actual location, the effective virtual address is used as the operand. For example, if execution of this instruction occurs with a zero indirect address bit and a zero in the index field, then the actual bit configuration in the address field of EAX is copied into bit positions 10-23 of the index register.

In the monitor mode, EAX copies the effective virtual address into bit positions 10-23 of the index register, as described above. However, if the effective address is subject to the user memory map (i.e., if bit position 0 of the EAX instruction or any indirect access contains a 1), bit 0 of the index register is set to 1; otherwise, bit 0 of the index register is reset to 0. Bits 1 through 9 of the index register are unaffected in the monitor mode.

Affected: $(X)_{0, 10-23}$ Timing: 2

XMA EXCHANGE MEMORY AND A

U	X	0	62	I	Reference address
0	1	2	3 8	9	10 23

XMA loads the effective word into the A register and, simultaneously, stores the contents of the A register in the effective location.

Affected: (A), (EL) Timing: 3

ARITHMETIC INSTRUCTIONS

ADD ADD

U	X	0	55	I	Reference address
0	1	2	3 8	9	10 23

ADD algebraically adds the effective word to the contents of the A register and loads the sum into the A register.

After execution of ADD, bit position 0 of the index (X) register contains the carry from bit position 0 of the 24-bit adder. Therefore, the programmer should be careful when attempting to hold a full word quantity in X while performing an ADD.

If both operands have the same sign but the sign of the sum is different, overflow has occurred, in which case the computer sets the overflow indicator; otherwise, the overflow indicator is unaffected.

Affected: (A), $(X)_0$, Of Timing: 2

ADC ADD WITH CARRY

U	X	0	57	I	Reference address
0	1	2	3 8	9	10 23

ADD WITH CARRY is used to perform multiprecision addition. Using the instruction ADD, the program adds the 24 low-order bits of the numbers (ADD automatically retains the carry in the sign position of the X register). Then, the program adds the next 24 bits of the numbers, using ADC, which also adds the carry bit (previously generated) into the low-order position of the adder. The program then continues with as many ADC instructions as are necessary to add the numbers.

After execution of ADC, bit position 0 of the index (X) register contains the carry from bit position 0 of the 24-bit adder. Therefore, the programmer should be careful when attempting to hold a full word quantity in X while performing an ADD WITH CARRY.

The computer automatically clears the overflow indicator prior to the execution of this instruction since overflow resulting from the subtraction of the low-order portions of the numbers is not meaningful.

If both operands have the same sign but the sign of the sum is different, an overflow has occurred, in which case, the computer sets the overflow indicator to 1; otherwise the computer resets the overflow indicator to 0.

Affected: (A), $(X)_0$, Of Timing: 2

Example:

Assume the A and B registers contain a double-precision number to which the double-precision number in locations M (15034166B) and N (12300000B) is to be added. The less significant halves of the numbers are in the B register and in location N. The program is:

Instruction	(A, B)	$(X)_0$
(Prior to execution)	20314624, 71510426B	-
XAB[†]	71510426, 20314624B	-
ADD N	04010426, 20314624B	1
XAB	20314624, 04010426B	1
ADC M	35351013, 04010426B	0

ADM ADD A TO MEMORY

U	X	0	63	I	Reference address
0	1	2	3 8	9	10 23

ADM adds the contents of the A register to the effective word and stores the result in the effective location.

If both operands have the same sign but the sign of the result is opposite, an overflow has occurred, in which case the computer sets the overflow indicator to 1; otherwise, the overflow indicator is unaffected.

Affected: (EL), Of Timing: 3

[†] XAB is the mnemonic for the instruction EXCHANGE A AND B (see "Register Change Instructions").

MIN MEMORY INCREMENT

U	X	0	61	I	Reference address
0	1	2	3 8	9	10 23

MIN adds 1 to the value of the effective word and stores the resulting sum in the effective location.

Overflow occurs with this instruction if and only if the effective word is 37777777B before execution, in which case 40000000B is the result in the effective location and the overflow indicator is set to 1. If no overflow occurs, the overflow indicator is unaffected.

Affected: (EL), Of Timing: 3

SUB SUBTRACT

U	X	0	54	I	Reference address
0	1	2	3 8	9	10 23

SUBTRACT inverts (forms the one's complement of) the effective word, adds the inverted word plus 1 to the contents of the A register, and loads the result into the A register.

After execution of SUB, bit position 0 of the index (X) register contains the carry from bit position 0 of the 24-bit adder. Therefore, the programmer should be careful when attempting to hold a full word quantity in X while performing a subtraction.

If the sign of the value in A is equal to the sign of the inverted word but the sign of the result is different, overflow has occurred, in which case, the computer sets the overflow indicator to 1; otherwise, the overflow indicator is unaffected.

Affected: (A), $(X)_0$, Of Timing: 2

SUC SUBTRACT WITH CARRY

U	X	0	56	I	Reference address
0	1	2	3 8	9	10 23

SUBTRACT WITH CARRY is used to perform multiple-precision subtractions. The program desiring to perform a multiple-precision subtraction first uses the SUBTRACT instruction to form the low-order 24 bits of the result. The SUB instruction automatically retains the carry bit in the sign position of the X register. The program then obtains the rest of the multiple-precision result by performing successive SUC instructions on the remaining portions of the multiple-precision numbers. The SUC instruction also automatically retains the carry bit in the sign position of the X register.

SUBTRACT WITH CARRY inverts (forms the one's complement of) the effective word, adds the inverted word plus the carry bit (sign position of the X register) to the contents of the A register, and loads the result into the A register.

After execution of SUC, bit position 0 of the index (X) register contains the carry from bit position 0 of the 24-bit adder. Therefore, the programmer should be careful when attempting to hold a full word quantity in X while performing a SUBTRACT WITH CARRY.

The computer automatically clears the overflow indicator prior to the execution of this instruction since overflow resulting from the subtraction of the low-order portions of the numbers is not meaningful.

If the sign of the value in A is equal to the sign of the inverted word but the sign of the result in A is opposite, overflow has occurred, in which case the computer sets the overflow indicator to 1; otherwise, the computer resets the overflow indicator to 0.

Affected: (A), $(X)_0$, Of Timing: 2

Example:

Assume that registers A and B and memory location M contain a triple-precision number from which the triple-precision number in locations L, L+1, and L+2 is subtracted.

(A, B, M): 36142070B, 31567000B, 10000001B

(L, L+1, L+2): 14236213B, 46120000B, 10000000B

The sign of one triple-precision number is in A_0, while its 71 binary digits are in A_{1-23}, B_{0-23}, and M_{0-23}. The sign of the other number is in L_0, and its 71 digits are in L_{1-23}, $L{+}1_{0-23}$, and $L{+}2_{0-23}$.

Execution:

Instruction	(A, B) after execution	$(X)_0$
XMA M	10000001, 31567000B	-
SUB L+2	00000001, 31567000B	1
XMA M	36142070, 31567000B	1
XAB	31567000, 36142070B	1
SUC L+1	63447000, 36142070B	0
XAB	36142070, 63447000B	0
SUC L	21704654, 63447000B	1

Answer: 21703654, 63447000, 00000001B

MUL MULTIPLY

U	X	0	64	I	Reference address
0	1	2	3 8	9	10 23

MULTIPLY multiplies the contents of the A register by the effective word and loads the fraction product into the A and B registers, with the more significant portion in A. The original contents of B do not affect the operation of the MULTIPLY instruction and are destroyed. The sign of the product is in A_0; the bit in B_0 is part of the product, not treated as a sign bit. Since the product contains at most 46 significant bits, the content of B_{23} is zero.

If the multiplier and multiplicand are both considered integers (i.e., with a binary point to the right of bit position 23), the binary point of the product is to the right of bit position 22 of the B register; thus, the entire result must be shifted 1 bit position to the right to obtain the correct integer product.

If the multiplier and multiplicand both have the value 40000000B, overflow occurs and the computer sets the overflow indicator to 1; otherwise, the overflow indicator is not affected.

Affected: (A), (B), Of   Timing: 4

Example, multiplication of 3 by 3:

		Before execution	After execution
(A, B)	=	00000003, xxxxxxxxB	00000000, 00000022B
EW	=	00000003B	00000003B

Note that

00000000, 00000011B scaled at 47

is equal to

00000000, 00000022B scaled at 46

DIV DIVIDE

U	X	0	65	I	Reference address
0	1	2	3 – 8	9	10 – 23

DIVIDE divides the contents of the A and B registers, treated as a double-precision number, by the effective word, loads the fractional quotient into the A register, and loads the fractional remainder into the B register.

During execution of the DIV instruction, the contents of the A and B registers (dividend) taken as a double-precision number are divided by the single-precision contents of the effective location (divisor). If the dividend is a single-precision number, the program should clear the B register prior to executing DIV, or erroneous results may occur. Although a double-length dividend is used, DIV is a single-precision operation; it should not be confused with a double-precision divide operation that uses a double-length divisor and produces a double-length quotient.

After execution of DIV, the single-precision quotient replaces the contents of the A register, and the remaining portion of the dividend that has not been divided (undivided remainder) replaces the contents of the B register. The quotient is signed in accordance with algebraic convention, that is, positive if dividend and divisor signs are alike, but negative otherwise. However, DIV generates only 23 magnitude bits and, if the magnitude of the quotient is so small as to require more than 23 bits to resolve, DIV may produce a zero quotient regardless of the required sign; but the remainder reflects the undivided portion of the original dividend. The binary scaling of the quotient is equal to the dividend scale factor minus the divisor scale factor.

The undivided remainder replaces the contents of the B register and has the same sign as the original dividend. It is scaled, in B, at dividend scaling minus 23.

No overflow occurs if $-1 \le \frac{(A, B)}{EW} < 1$ (if the quotient is greater than or equal to minus one but strictly less than plus one). If the quotient exceeds these boundaries, overflow occurs and the computer sets the overflow indicator to 1. In this latter case, the results are not arithmetically correct.

Affected: (A, B), Of   Timing: 10

Example 1:

		Before execution	After execution
(A, B)	=	00000000, 00000016B	00000002, 00000001B
EW	=	00000003B	00000003B
Of	=	x	x

Example 2:

		Before execution	After execution
(A, B)	=	37777777, 00000002B	40000000, 00000001B
EW	=	44433343B	44433343B
Of	=	x	1

LOGICAL INSTRUCTIONS

ETR EXTRACT

U	X	0	14	I	Reference address
0	1	2	3 – 8	9	10 – 23

ETR performs a logical AND between corresponding bits of the A register and the effective word and loads the result into A. This instruction performs the operation (bit by corresponding bit) according to the following table:

A_i	EW_i	Result in A_i
0	0	0
0	1	0
1	0	0
1	1	1

Affected: (A)   Timing: 2

Example:

		Before execution	After execution
(A)	=	64231567B	00231400B
EW	=	00777600B	00777600B

MRG MERGE

U	X	0	16	I	Reference address
0	1	2	3 – 8	9	10 – 23

MRG performs a logical inclusive OR between corresponding bits of the A register and the effective word and loads the result into A. This instruction performs the operation (bit by corresponding bit) according to the following table:

A_i	EW_i	Result in A_i
0	0	0
0	1	1
1	0	1
1	1	1

Affected: (A)   Timing: 2

Example:

		Before execution	After execution
(A)	=	06446254B	06746756B
EW	=	02340712B	02340712B

EOR EXCLUSIVE OR

U	X	0	17	I	Reference address
0	1	2	3–8	9	10–23

EOR performs a logical exclusive OR between corresponding bits of the A register and the effective word and loads the result into A. This instruction performs the operation (bit by corresponding bit) according to the following table:

A_i	EW_i	Result in A_i
0	0	0
0	1	1
1	0	1
1	1	0

Affected: (A) Timing: 2

Example:

		Before execution	After execution
(A)	=	34165031B	44112010B
EW	=	70077021B	70077021B

The proper memory word configuration logically inverts selected bit positions of the A register. If the effective word is 77777777B, a one's complement of A results.

REGISTER CHANGE INSTRUCTIONS

The facility to operate on and exchange data between the A, B, and index registers is available within the set of micro-instructions in the register change group.

All instructions in the group use the same operation code, 46B. Bit positions 1 and 14 through 23 of the address field specify the function to be performed by each micro-instruction. The programmer may specify combinations of address bits to perform simultaneous operations.

If the selected bits specify that the computer copy two registers into a third during one operation, a merge of the former two registers into the latter results. If the selected control bits specify that the computer copy into a register and clear that same register, the clear operation has no effect. The function of each address bit is:

Bit	Function
1	Clear X
14	Copy -(A) into A
15	Copy (A) into X
16	Copy (X) into A
17	Bits 15-23 only†
18	Copy (X) into B
19	Copy (B) into X
20	Copy (B) into A
21	Copy (A) into B
22	Clear B
23	Clear A

†See STORE EXPONENT, LOAD EXPONENT, and EXCHANGE EXPONENTS

Indirect addressing and indexing do not apply to these instructions.

These instructions require one machine cycle regardless of the number of functions performed. As an aid to the programmer, the most useful combinations have mnemonic designations assigned to them that are recognized by standard XDS 940 programming systems.

CLA CLEAR A

0	46	00001
0–2	3–8	9–23

CLA clears the contents of the A register to zero.

Affected: (A) Timing: 1

CLB CLEAR B

0	46	00002
0–2	3–8	9–23

CLB clears the contents of the B register to zero.

Affected: (B) Timing: 1

CLAB CLEAR AB

0	46	00003
0–2	3–8	9–23

CLAB clears the contents of both the A and B registers to zero.

Affected: (A), (B) Timing: 1

CLX CLEAR INDEX

2	46	00000
0–2	3–8	9–23

CLX clears the contents of the index (X) register to zero.

Affected: (X) Timing: 1

CLEAR CLEAR A, B, AND X

2	46	00003
0–2	3–8	9–23

CLEAR clears the contents of the A, B, and index (X) registers to zero.

Affected: (A), (B), (X) Timing: 1

CAB COPY A INTO B

0	46	00004
0–2	3–8	9–23

CAB copies the contents of the A register into the B register.

Affected: (B) Timing: 1

CBA COPY B INTO A

0	46	00010
0 2	3 8	9 23

CBA copies the contents of the B register into the A register.

Affected: (A) Timing: 1

XAB EXCHANGE A AND B

0	46	00014
0 2	3 8	9 23

XAB copies the contents of the A register into the B register and, simultaneously, copies the contents of the B register into the A register.

Affected: (A), (B) Timing: 1

ABC COPY A INTO B, CLEAR A

0	46	00005
0 2	3 8	9 23

ABC copies the contents of the A register into the B register and then clears the A register to zero.

Affected: (A), (B) Timing: 1

BAC COPY B INTO A, CLEAR B

0	46	00012
0 2	3 8	9 23

BAC copies the contents of the B register into the A register and then clears the B register to zero.

Affected: (A), (B) Timing: 1

CAX COPY A INTO INDEX

0	46	00400
0 2	3 8	9 23

CAX copies the contents of the A register into the index register.

Affected: (X) Timing: 1

CXA COPY INDEX INTO A

0	46	00200
0 2	3 8	9 23

CXA copies the contents of the index register into the A register.

Affected: (A) Timing: 1

XXA EXCHANGE INDEX AND A

0	46	00600
0 2	3 8	9 23

XXA copies the contents of the index register into the A register and, simultaneously, copies the contents of the A register into the index register.

Affected: (A), (X) Timing: 1

CBX COPY B INTO INDEX

0	46	00020
0 2	3 8	9 23

CBX copies the contents of the B register into the index register.

Affected: (X) Timing: 1

CXB COPY INDEX INTO B

0	46	00040
0 2	3 8	9 23

CXB copies the contents of the index register into the B register.

Affected: (B) Timing: 1

XXB EXCHANGE INDEX AND B

0	46	00060
0 2	3 8	9 23

XXB copies the contents of the index register into the B register and, simultaneously, copies the contents of the B register into the index register.

Affected: (B), (X) Timing: 1

STE STORE EXPONENT

0	46	00122
0 2	3 8	9 23

STE copies the 9 least significant bits of the B register into the 9 least significant bit positions of the index register, extends bit 15 of the index register (the sign of the exponent) into bit position 0 of the index register, and then clears the 9 least significant bit positions of B.

Affected: $(B)_{15-23}$, (X) Timing: 1

Example:

	Before execution	After execution
(B) =	64152713B	64152000B
(Index) =	---	77777713B

LDE LOAD EXPONENT

0	46	00140
0–2	3–8	9–23

LDE copies the 9 least significant bits of the index register into the 9 least significant bit positions of the B register. The 9 least significant bit positions of B are cleared prior to the transfer.

Affected: $(B)_{15-23}$ Timing: 1

Example:

	Before execution	After execution
(B) =	34765712B	34765151B
(Index) =	00000151B	00000151B

XEE EXCHANGE EXPONENTS

0	46	00160
0–2	3–8	9–23

XEE exchanges the 9 least significant bits of the B register with the 9 least significant bits of the index register. The exchange loses no information. The new bit 15 of the index register (the sign of the exponent) is then extended into bit position 0.

Affected: $(B)_{15-23}$, (X) Timing: 1

Example:

	Before execution	After execution
(B) =	67142355B	67142133B
(Index) =	77777133B	00000355B

CNA COPY NEGATIVE INTO A

0	46	10000
0–2	3–8	9–23

CNA copies the two's complement of the contents of the A register into the A register.

Affected: (A) Timing: 1

BRANCH INSTRUCTIONS

Branch instructions conditionally or unconditionally change the course of the program by altering the contents of the program counter. The programmer should note that these instructions branch to locations determined by the effective address; this means that the branch can operate with all levels of indirect and indexed addressing.

BRU BRANCH UNCONDITIONALLY

U	X	0	01	I	Reference address
0	1	2	3–8	9	10–23

BRU takes the next instruction from the location determined by the effective address.

When the computer is in the normal mode, a BRU instruction with the indirect address bit set to 1 clears the highest-priority active interrupt level, in addition to branching to the effective location. However, when the computer is in the monitor mode or the user mode, no interrupt level is affected.

Affected: (P), highest-priority active interrupt level Timing: 1

BRX INCREMENT INDEX AND BRANCH

U	X	0	41	I	Reference address
0	1	2	3–8	9	10–23

BRX adds 1 to the contents of the index register. If the resultant index register value contains a 1 in bit position 9, the computer transfers control to the effective location. If not, it takes the next instruction in sequence.

If a BRX instruction is indexed, any transfer of control is to the effective address determined by the value of the index immediately prior to the execution of BRX. The test for transfer is on the incremented value of the index register, just as if the BRX instruction were not indexed.

The 9 most significant bits of the index register (bits 0-8) have no effect on the execution of the instruction, but may be affected by it.

If a branch occurs in the normal mode, an interrupt cannot occur following the execution of this instruction; however, an interrupt can occur following this instruction in the user and monitor modes.

Affected: (X), (P) Timing: 1, if branch; 2, if no branch

Example:

Location	Instruction	(X Register)
0777B	STA 1500B	77777776B
1000B	BRX 1006B	77777777B
1001B	LDA 2000B	
⋮	⋮	
1006B	BRX 1001B	00000000B
1007B	LDA 2100B	00000000B

The execution of these instructions is in the following order as given by their locations:

0777B
1000B
1006B
1007B

BRM MARK PLACE AND BRANCH

0	X	0	43	I	Reference address
0	1	2	3–8	9	10–23

In the normal and user modes, MARK PLACE AND BRANCH performs the following operations:

1. stores the state of the overflow indicator in bit position 0 of the effective location

2. stores the state of the memory extension registers (EM3 and EM2) in bit positions 3-5 and 6-8, respectively, of the effective location
3. resets bits 1, 2, and 9 of the effective location to 0's
4. stores the contents of the P register (the address of the BRM instruction) in bit positions 10-23 of the effective location

In the monitor mode, BRM performs the following operations:

1. stores a mode indicator in bit position 0 of the effective location (i.e., bit 0 is set to 1 if the previous instruction was executed in the user mode and bit 0 is reset to 0 if the previous instruction was executed in the normal or monitor mode). This action allows for a return from a subroutine independent of the computer mode at the time the subroutine was entered with the BRM instruction.
2. stores the state of the overflow indicator in bit position 2 of the effective location
3. stores the state of the memory extension registers (EM3 and EM2) in bit positions 3-5 and 6-8, respectively, of the effective location
4. resets bit positions 1 and 9 of the effective location to zero
5. stores the contents of the P register in bit positions 10-23 of the effective location

Regardless of the computer mode, BRM loads the value of the effective address plus 1 into the P register; thus, the next instruction is taken from the next location after effective location. If the BRM instruction is executed as the operand of an EXECUTE instruction (see page 30), the stored P register value is the address of the initial EXECUTE instruction rather than the address of the BRM instruction.

BRM is used to enter subroutines where a return to the main program is desired after completing the subroutine. The subroutine can return program control to the main program by executing a BRU indirect, a BRR, or a BRI instruction depending upon the mode of the computer and upon the conditions under which the BRM was executed.

Affected: (EL), (P) Timing: 2

Example: BRM 1517B

		Before execution	After execution
(P)	=	522B	1520B
(EM3)	=	3	3
(EM2)	=	2	2
(Of)	=	1	1
(1517B)	=	xxxxxxxxB	53200522B
Mode	=	user	user

BRR RETURN BRANCH

0	X	0	51	I	Reference address
0	1	2	3–8	9	10–23

In the normal and user modes, RETURN BRANCH performs a logical OR between bit 0 of the effective word and the overflow indicator, places the result in the overflow indicator, and then loads the P register with a value equal to 1 plus the contents of bit positions 10-23 of the effective location.

In the monitor mode, BRR performs a logical OR between bit 2 of the effective word and the overflow indicator, places the result in the overflow indicator, and then loads the P register with a value equal to 1 plus the contents of bit positions 10-23 of the effective word. If bit 0 of the effective word is a 1 (or a 1 is detected in bit position 0 of an indirect access), the computer automatically enters the user mode before the next instruction is accessed. If bit 0 of the effective word is a 0 (and no 1 is detected in bit position 0 of an indirect access), the computer remains in the monitor mode.

Affected: Of, (P) Timing: 2

Example: BRR 1517B

		Before execution	After execution
(P)	=	1540B	523B
(EM3)	=	3	3
(EM2)	=	2	2
(Of)	=	0	1
(1517B)	=	53200522B	53200522B
Mode	=	monitor	user

BRI BRANCH AND RETURN FROM INTERRUPT ROUTINE (940 only, Privileged)

0	X	0	11	I	Reference address
0	1	2	3–8	9	10–23

In the normal mode, BRI performs the following:

1. loads bit 0 of the effective word into the overflow indicator
2. loads the P register with bits 10-23 of the effective word
3. clears the highest-priority active interrupt level

In the monitor mode, BRI performs the following:

1. loads bit 2 of the effective word into the overflow indicator
2. loads the P register with bits 10-23 of the effective word
3. clears the highest-priority active interrupt level
4. if bit 0 of the effective word is a 1 (or a 1 is detected in bit position 0 of an indirect access), the computer automatically enters the user mode before the next instruction is accessed; if bit 0 of the effective word is a 0 (and no 1 is detected in bit position 0 of an indirect access), the computer remains in the monitor mode

If execution of BRI is attempted while the computer is in the user mode, the BRI instruction is not executed. Instead, the computer traps to location 40B with the P register containing the address of the BRI instruction. If execution of the BRI instruction was attempted as the operand of an EXECUTE instruction, the P register contains the address of the initial EXECUTE instruction rather than the address of the BRI instruction.

Affected: Of, (P), highest-priority active interrupt level Timing: 2

Example: BRI 2000B

		Before execution	After execution
(P)	=	2017B	2537B
(Of)	=	0	1
(2000B)	=	53202537B	53202537B
Mode	=	monitor	user

TEST AND SKIP INSTRUCTIONS

SKE SKIP IF A EQUALS MEMORY

U	X	0	50	I	Reference address
0	1	2	3 – 8	9	10 – 23

SKE compares the contents of the A register with the effective word. If the contents of A equal the effective word, the computer skips the next instruction in sequence and executes the following instruction. If the contents of A do not equal the effective word, the computer executes the next instruction in sequence.

Affected: (P) Timing: 2, if no skip
3, if skip

SKG SKIP IF A GREATER THAN MEMORY

U	X	0	73	I	Reference address
0	1	2	3 – 8	9	10 – 23

SKG algebraically compares the contents of the A register with the effective word. If the contents of A are greater than the effective word, the computer skips the next instruction in sequence and executes the following instruction. If the contents of A are less than or equal to the effective word, the computer executes the next instruction in sequence.

Affected: (P) Timing: 2, if no skip
3, if skip

SKM SKIP IF A EQUALS MEMORY ON B MASK

U	X	0	70	I	Reference address
0	1	2	3 – 8	9	10 – 23

SKM compares selected bits of the A register with corresponding bits of the effective word. If the selected bits in A are all identical to corresponding bits of the effective word, the computer skips the next instruction in sequence and executes the following instruction. If the selected bits in the A register are not all identical to corresponding bits of the effective word, the computer executes the next instruction in sequence.

The programmer selects the bits in A to be compared by placing ones in the corresponding bit positions of the B register and zeros in the remaining bit positions of B.

SKM treats the contents of A, B, and the effective location to be unsigned, 24-bit, nonnumeric quantities, and does not alter them.

Affected: (P) Timing: 2, if no skip
3, if skip

Example:

(A)	(B)	(EL)
00043007B	00177000B	57643240B

Since SKM compares bit positions 8-14 only (as determined by B), and (A) = (EL) in these positions, a skip occurs. Note that if (B) = 0, a skip occurs regardless of (A) and (EL). Note also that if (B) = 77777777B, the operation of SKM is identical to that of the instruction SKE.

SKA SKIP IF A AND MEMORY DO NOT COMPARE ONES

U	X	0	72	I	Reference address
0	1	2	3 – 8	9	10 – 23

SKA compares the contents of the A register, bit by bit, with the effective word. If the contents of the A register and the effective word do not have ones in any corresponding bit positions, the computer skips the next instruction in sequence and executes the following instruction. If the contents of the A register and the effective word do have ones in at least one corresponding bit position, the computer executes the next instruction in sequence.

The instruction logically ANDs corresponding bits in A and the effective word, based on the following table:

(A)	EW	Result
0	0	0
0	1	0
1	0	0
1	1	1

If the result produces a 1 in any bit position, a skip does not occur.

Note: Different configurations of the effective word result in a wide variety of conditional operations for use by the programmer. Some representative configurations are:

Effective word configuration	Operation
40000000B	Skip if (A) is positive
77777777B	Skip if (A) = 0
00000001B	Skip if (A) is even

Contents of A register	
40000000B	Skip if effective word is positive
77777777B	Skip if effective word = 0
00000001B	Skip if effective word is even

Affected: (P) Timing: 2, if no skip
3, if skip

SKB SKIP IF B AND MEMORY DO NOT COMPARE ONES

U	X	0	52	I	Reference address
0	1	2	3 – 8	9	10 – 23

The operation of SKB is identical to that of SKA, but uses the contents of the B register instead of the contents of the A register.

Affected: (P) Timing: 2, if no skip
3, if skip

SKN SKIP IF MEMORY NEGATIVE

U	X	0	53	I	Reference address
0	1	2	3 – 8	9	10 – 23

If the effective word is a negative value (i.e., bit 0 of the effective word is a 1), the computer skips the next instruction in sequence and executes the following instruction. If the effective word is a positive or zero value, the computer executes the next instruction in sequence.

Affected: (P) Timing: 2, if no skip
3, if skip

SKR REDUCE MEMORY, SKIP IF NEGATIVE

U	X	0	60	I	Reference address
0	1	2	3 – 8	9	10 – 23

SKR reduces the value of the effective word by one, places the result in the same location, and then tests the effective word for being a negative value. If the effective word is a negative value after being reduced, the computer skips the next instruction in sequence and executes the following instruction. If the effective word is a positive or zero value after being reduced, the computer executes the next instruction in sequence.

An overflow occurs if the initial value of the effective word is 40000000B, in which case the resulting effective word is 37777777B, and the overflow indicator is set. If no overflow occurs, the overflow indicator is unaffected.

Affected: (EL), Of, (P) Timing: 3

SKD DIFFERENCE EXPONENTS AND SKIP

U	X	0	74	I	Reference address
0	1	2	3 – 8	9	10 – 23

SKD subtracts bits 15 through 23 of the effective word from bits 15 through 23 of the B register, and stores the absolute magnitude of the difference in the X register. If the 9 low-order bits of the effective word are less than or equal to the 9 low-order bits of the B register, the computer executes the next instruction in sequence; otherwise, the computer skips the next instruction in sequence and executes the following instruction.

Affected: $(X)_{15-23}$ Timing: 2, if no skip
3, if skip

SHIFT INSTRUCTIONS

The shift instructions operate on the contents of the A and B registers and offer a complete facility for right and left shifting, cycling, and normalizing the contents of these two registers. The A and B registers, in combination, form a double-length register whose double-length contents can be shifted, cycled, or normalized. This double-length register is named "AB".

When the contents of the AB register shift right, bits from bit position 23 of the A register shift into bit position 0 of the B register. When the AB register shifts left, bits from bit position 0 of the B register shift into bit position 23 of the A register.

The 48-bit contents of the AB register may be cycled using the shift instructions. When the contents of the AB register cycle, the bits that shift from one end of the one register copy into the other end of the other register.

These instructions use the instruction code to determine the direction of shift (66 = right; 67 = left); bits 10-11 (octal position 3) of the instruction address determine the method of shifting as follows:

Bits 10, 11	Function
00	AB shift
10	AB cycle
01	Normalize (left only)

Since the type of shift and number of shifts are determined by bits 10 through 23 of the effective virtual address, indirect addressing and indexing drastically alter the action specified in a shift instruction. When computing the effective virtual address for a shift instruction,

14-bit indexing is performed with all indirectly addressed operands, and

9-bit indexing is performed with all directly addressed operands.

That is, indexing with a direct address can affect only the 9-bit shift count.

When the computer decodes a shift instruction, bit positions 15 through 23 of the effective address of the instruction determine the amount of the shift. The computer treats these nine bits as an unsigned count. If the initial count is equal to zero, no shifting occurs. If the initial count is greater than 48, it is set to 48 prior to shifting. Once the shift begins, the count is reduced by 1 for each position shifted, until it reaches zero. The count C in the following instructions indicates the number of places to be shifted. Shift timing is:

Left shift and normalize count	Cycles	Right shift count
0 - 6	2	0 - 3
7 - 26	3	4 - 14
27 - 46	4	15 - 25
47 - 48	5	26 - 36
	6	37 - 47
	7	48

RSH RIGHT SHIFT AB

U	X	0	66	I	0 0 0 0 0	C
0	1	2	3–8	9	10–14	15–23

RSH shifts the contents of the AB register (that is, A and B registers) right the number of places specified by bits 15 through 23 of the effective address. The bit in the sign position of A does not shift, but its value is copied into the vacated bit positions of the shifted number. The bit in the sign position of B is shifted as a magnitude bit. Bits shifted out of A_{23} shift into B_0. Bits shifting past B_{23} are lost.

Affected: (AB) Timing: 2-7

Example:

The instruction is: RSH 18

	Before execution	After execution
(A, B) =	45261237, 27651260B	77777745, 26123727B

Note: This instruction may be used to perform scaling of floating-point numbers by use of indexing, where the difference of the exponents is in the index register as a positive quantity.

LRSH LOGICAL RIGHT SHIFT AB

0	X	0	66	24	C
0	1	2	3–8	9–14	15–23

LRSH shifts the contents of AB right the number of places specified by bits 15 through 23 of the effective address. The bits in the sign position of A and the sign position of B shift with the rest of the number. Vacated bit positions on the left are filled with zeros. Bits shifting out of A_{23} shift into B_0. Bits shifting past B_{23} are lost.

Affected: (AB) Timing: 2-7

RCY RIGHT CYCLE AB

0	X	0	66	20	C
0	1	2	3–8	9–14	15–23

RCY shifts the contents of the AB register right the number of places specified in bits 15 through 23 of the effective address. The bits in the sign positions of A and B shift like any other bits in the number. Bits shifting out of A_{23} shift into B_0. Bits shifting out of B_{23} shift into A_0. The computer treats the double-length register as if it were circular and cycles it onto itself; it loses no bits.

Affected: (AB) Timing: 2-7

Example:

The instruction is: RCY 15

	Before execution	After execution
(A, B) =	61235703, 41537701B	37701612, 35703415B

LSH LEFT SHIFT AB

U	X	0	67	I	0 0 0 0 0	C
0	1	2	3–8	9	10–14	15–23

LSH shifts the contents of the AB register left the number of places specified in bits 15 through 23 of the effective address. Bits shift left through the sign position of A, but when a bit, different in value from the original sign, shifts into the sign position, the computer sets the overflow indicator. Bits shifting out of B_0 shift into A_{23}. Bits shifting past position 0 in A are lost. Zeros fill the vacated bit positions on the right end of the B register.

Affected: (AB), Of Timing: 2-5

Example:

The instruction is: LSH 18

	Before execution	After execution
(A, B) =	46712370, 64132711B	70641327, 11000000B

LCY LEFT CYCLE AB

0	X	0	67	20	C
0	1	2	3–8	9–14	15–23

LCY shifts the contents of the AB register left the number of places specified in bits 15 through 23 of the effective address. The bits in the sign positions of A and B shift like any other bits in the number. Bits shifting out of B_0 shift into A_{23}. The instruction copies bits that shift from bit position 0 of A into bit position 23 of B. The computer treats the double-length register as if it were circular and cycles it onto itself; it loses no bits.

Affected: (A, B) Timing: 2-5

Example:

The instruction is: LCY 9

	Before execution	After execution
(A, B) =	71432560, 34156723B	32560341, 56723714B

NOD NORMALIZE AND DECREMENT X

0	X	0	67	10	C
0	1	2	3–8	9–14	15–23

NOD shifts the contents of the AB register left until (1) a bit appears in position 1 of A that is not equal to the bit in the sign position of A, or (2) until C shifts occur. The computer keeps count of the number of places shifted and when the normalize operation is completed, it subtracts the count from the contents of the index register and places the result back into the index. If, in the attempt to normalize, shifting exceeds 48 places, the contents of the AB register were initially zero. In this case, the computer subtracts 48 from the index register. Zeros fill the vacated positions.

The number C, placed in address bit positions 15 through 23, is an upper limit for the number of left shifts that will occur. The programmer must ensure that C is sufficiently large to permit a complete normalization.

Affected: (A, B), (X) Timing: 2-5

Example:

The instruction is: NOD 30

	Before execution	After execution
(A, B) =	00004632, 76124035B	23153705, 20164000B
(X) =	00000000B	77777765B

CONTROL INSTRUCTIONS

HLT HALT (Privileged)

0	00	00000
0 2	3 8	9 23

When the computer executes this instruction, it halts computation and lights the HALT indicator in the console. Before halting, the computer increments the P register and brings the next instruction to the C register to be displayed. Also, if an interlaced I/O operation is in progress, it continues to completion.

The computer turns off the HALT indicator and continues to the next instruction if either of the following conditions occurs:

1. the RUN-IDLE-STEP switch is moved to IDLE and then to RUN or STEP
2. an interrupt level advances to the active state while the RUN-IDLE-STEP switch is in RUN (in this case, the interrupt-servicing routine is processed before the instruction after HALT is executed)

If execution of HLT is attempted while the computer is in the user mode, the HLT instruction is not executed. Instead, the computer traps to location 40B with the P register containing the address of the HLT instruction. If execution of the HLT instruction was attempted as the operand of an EXECUTE instruction, the P register contains the address of the initial EXECUTE instruction rather than the address of the HLT instruction.

Affected: HALT indicator Timing: 1 + wait

NOP NO OPERATION

0	20	00000
0 1 2	3 8	9 14 15 23

Executing NOP does not affect the A register, B register, X register, or memory. Indirect addressing and indexing do not apply to this instruction.

Affected: None Timing: 1

EXU EXECUTE

U	X	0	23	I	Reference address
0	1	2	3 8	9	10 23

EXU causes the effective word to be executed as an instruction without altering the contents of the program counter. If the effective word is not a branch, skip, or another EXECUTE instruction, the computer executes the next instruction, after it executes the effective word.

If the effective word is a branch instruction, program control goes to the effective address of the branch and not to the next instruction in sequence following the EXECUTE instruction.

If the effective word is a skip instruction, then, depending on the skip decision, program control returns to the next instruction, or the next instruction plus one, following the EXECUTE instruction.

If the effective word is another EXECUTE instruction, the above process continues identically, with the normal return being the location of the initial EXECUTE instruction plus one. This process can cascade indefinitely, but can be interrupted in the user and monitor modes (see Section 2, "Hardware Hang-up Prevention").

Affected: Determined by executed instruction Timing: 1 + executed instruction

BREAKPOINT TESTS (Privileged)

Four configurations of the SKIP IF SIGNAL NOT SET (SKS) instruction test the status of the BREAKPOINT switches on the computer control panel singly or in any combination. If any one of the tested BREAKPOINT switches is reset, the computer skips the next location in sequence and executes the following instruction. If none of the tested BREAKPOINT switches is reset, the computer executes the next instruction sequence.

Mnemonic	Name of Instruction	Octal Configuration
BPT1	BREAKPOINT 1 TEST	0 40 20400
BPT2	BREAKPOINT 2 TEST	0 40 20200
BPT3	BREAKPOINT 3 TEST	0 40 20100
BPT4	BREAKPOINT 4 TEST	0 40 20040

If execution of BPT is attempted while the computer is in the user mode, the BPT instruction is not executed. Instead, the computer traps to location 40B with the P register containing the address of the BPT instruction. If execution of the BPT instruction was attempted as the operand of an EXECUTE instruction, the P register contains the address of the initial EXECUTE instruction rather than the address of the BPT instruction.

Affected: (P) Timing: 1, if no skip; 2, if skip

OVERFLOW INSTRUCTIONS

OVT OVERFLOW INDICATOR TEST AND RESET

0	22	00101
0 2	3 8	9 14 15 23

This instruction tests the status of the overflow indicator, skips or not accordingly, and turns the indicator off. If the indicator is off, the computer skips the next instruction in sequence and executes the following instruction. If the indicator is on, the computer turns the indicator off and then executes the next instruction in sequence.

In the normal and monitor modes, the instruction SKS 20001B may be used to test and reset the overflow indicator.

Affected: (P), Of　　　　Timing: 1, if no skip
2, if skip

OTO OVERFLOW INDICATOR TEST ONLY (940 only)

0	22	00100
0 1 2	3 8	9 14 15 23

This instruction tests (but does not change) the status of the overflow indicator. If the overflow indicator is on, the computer executes the next instruction in sequence; however, if the overflow indicator is off, the computer skips the next instruction in sequence and executes the following instruction.

Affected: (P)　　　　Timing: 1, if no skip
2, if skip

REO RECORD EXPONENT OVERFLOW

0	22	00010
0 2	3 8	9 23

This instruction causes the overflow indicator to be turned on if the content of bit 14 of the index register is not equal to the content of bit 15 of the index register; otherwise, the overflow indicator is not affected.

In the normal and monitor modes, the instruction EOM 20100B may be used to record exponent overflow.

Affected: Of　　　　Timing: 1

ROV RESET OVERFLOW INDICATOR

0	22	00001
0 2	3 8	9 23

ROV unconditionally resets (turns off) the overflow indicator.

In the normal and monitor modes, EOM 20001B may be used to reset the overflow indicator.

Affected: Of　　　　Timing: 1

MEMORY EXTENSION INSTRUCTIONS (Privileged)

SET EXTENSION REGISTER (Privileged)

0	06	2	0 0 0 0	S3	S2	R3	R2
0 2	3 8	9 11	12 15	16	17	18 20	21 23

This instruction sets (loads) memory extension registers 3 and/or 2 with the contents of fields R3 and R2, respectively. If S3 (bit 16) is a 1, the computer sets the contents of R3 (bits 18 through 20) into EM3. This destroys the previous contents of EM3. If S3 is a 0, the instruction does not affect EM3. If S2 (bit 17) is a 1, the computer sets the contents of R2 (bits 21 through 23) into EM2. This destroys the previous contents of EM2. If S2 is a 0, the instruction does not affect EM2. If both S3 and S2 are 1, the instruction loads both EM3 and EM2 simultaneously. If both S3 and S2 are 0, the instruction is effectively a "no-operation" instruction.

If execution of the instruction is attempted while the computer is in the user mode, the instruction is not executed. Instead, the computer traps to location 40B with the P register containing the address of the instruction. If execution of the SET EXTENSION REGISTER instruction was attempted as the operand of EXECUTE, the P register contains the address of the initial EXECUTE instruction rather than that of the SET EXTENSION REGISTER instruction.

Affected: (EM3), (EM2)　　　　Timing: 1

EXTENSION REGISTER TEST (Privileged)

0	40	4000	0	T
0 2	3 8	9 21	22	23

This instruction tests the extension registers as follows:

<u>T</u> <u>Test</u>

0 No test. The computer executes the next instruction in sequence.

1 Test EM2. If (EM2) ≠ 2, the computer skips the next instruction in sequence. If (EM2) = 2, the computer executes the next instruction in sequence.

2 Test EM3. If (EM3) ≠ 3, the computer skips the next instruction in sequence. If (EM3) = 3, the computer executes the next instruction in sequence.

3 Test EM3 and EM2. If (EM3) ≠ 3 <u>or</u> (EM2) ≠ 2, the computer skips the next instruction in sequence. When (EM3) = 3 <u>and</u> (EM2) = 2, the computer executes the next instruction in sequence.

If execution of the test instruction is attempted while the computer is in the user mode, the test instruction is not executed. Instead, the computer traps to location 40B with the P register containing the address of the test instruction. If execution of the test instruction was attempted as the operand of an EXECUTE instruction, the P register contains the address of the initial EXECUTE instruction rather than the address of the test instruction.

Affected: (P)　　　　Timing: 1, if no skip
2, if skip

4. INPUT/OUTPUT SYSTEM

The XDS 940 has a flexible input/output system to complement its high internal processing speed and versatile instruction repertoire. The system can transmit data in word, character, or single-bit form to and from the computer at the speed of internal computation. The input/output system assumes control of conditions imposed by individual characteristics of a wide variety of devices, yet it leaves a high degree of input/output control to the programmer.

The I/O system provides for the following kinds of input and output:

1. Input/output of data words, each one of which is under direct control of the program
2. Communication channel input/output of characters or words, time-shared with normal accesses to memory and multiplexed with computation
3. Communication channel input/output of characters or words, fully buffered and simultaneous with computation
4. Direct parallel input/output of up to 24 bits of information to and from external equipment, completely controlled and sequenced externally from the central processor
5. Direct parallel input/output of up to 24 bits of information to and from external registers under program control
6. Single-bit input/output, such as equipment on/off status, sense switches, and pulsing and sensing of special devices

COMMUNICATION CHANNEL INPUT/OUTPUT

XDS 940 communication channels provide fully buffered input/output control and transmission, multiplexed or simultaneous with computation. Up to eight data channels can be connected to the XDS 940, all operating independently of each other.

Each channel can control as many as 30 input/output devices and automatically handles character/word assembly and disassembly, input/output parity detection and generation, data transmission to and from memory, and end-of-transmission detection.

All channels are bidirectional and can communicate with 6-bit character devices or word devices of up to 24 bits. In the case of character-oriented devices, the number of characters to be contained in each word during the transmission is specified by program when the channel operation is initiated.

Each channel consists of a channel buffer and a channel interlace. The channel buffer assembles and disassembles data words as they are transmitted between core memory and the peripheral equipment. The channel interlace controls the transmission of blocks of data.

TIME-MULTIPLEXED COMMUNICATION CHANNELS

The XDS 940 includes as standard equipment one time-multiplexed communication channel (TMCC), with provision for addition of three additional channels. These channels are capable of automatically controlling the flow of data to and from memory at rates up to one word every 3.5 microseconds. These channels run independently of the central processor and only interfere with it to transfer data to or from memory.

The time-multiplexed channels use the memory access logic of the central processor to facilitate input and output of data words. The transfer of each word between a time-multiplexed channel and core memory requires two memory cycles. During this time, computation is delayed in the central processor. Priority for the use of the input/output logic is in the order: channel D, C, Y, W, with channel D having the highest priority. Any time-multiplexed channel operating with automatic interlace has priority over the central processor for memory access.

DIRECT MEMORY ACCESS SYSTEM

In addition to the time-multiplexed channels, a direct memory access system is included in the 940. This system uses a separate path to memory from those used by the central processor and the time-multiplexed communication channels. The separate path to memory allows data transfer through the direct access system without interfering with the central processor if the memory access is to a module that is not being addressed by the central processor. One to four direct access communication channels (DACC) can be attached to the direct access system. These channels operate like time-multiplexed channels, except that they are faster and provide for a true overlap of input/output with processing.

Each direct access channel has its own independent memory access logic. When a memory access is required to obtain or store a data word, computation is delayed one cycle if the access is in the same memory module being addressed by the central processor; if the module is not being addressed by the central processor, no time is lost and computation is unaffected. When two or more direct access channels require memory access simultaneously, priority is determined as described in "Channel Memory Access Priority", at the end of this section.

Transmissions between direct access channels and core memory are under the control of the channel. At the onset of each memory cycle, the control unit interrogates all direct access channels to determine if one of them requires a transfer to or from computer memory. If such is the case, the the computer connects the specified memory module to the selected direct access channel. If, simultaneously, the computer requires access to the same memory module, the computer requirement takes precedence over the channel and the data I/O is delayed one memory cycle unless an I/O data overrun (character rate error) is imminent; in which case the channel requirement takes precedence and computation is delayed one memory cycle. If the computer and a direct access channel are not accessing the same memory module, the transfer takes place without affecting computation speed. Thus, internal computation and direct access channel transmissions occur simultaneously and independently when the computer and the channel are accessing different memory modules. Channel control logic permits the transfer of only one word per memory cycle to and from the computer memory independent of the number of operating channels connected to the computer. Thus the maximum transfer rate for direct access is one word every memory cycle (approximately equal to 571,000 words per second, or in excess of two million characters per second).

The memory mapping system of the 940 Computer partitions core memory into blocks of 2048 words. Although core memory appears contiguous to a program of more than 2048 words, the actual memory space allocated to the program may be in noncontiguous blocks (see Section 2, "Memory Access Control"). However, the memory addresses for input/output are not controlled by the memory mapping system; thus, a data transfer of more than one word may overlap a virtual memory block boundary. The 940 direct access communication channels incorporate a provision to control the actual memory addresses used in an input/output operation so that such transitions across virtual memory block boundaries can be properly accomplished (with a minimum of programming effort and interrupt response requirements) when the actual memory blocks are noncontiguous.

A data multiplexing system, which uses the direct access memory connection, is also available as an option. This system consists of a data multiplex channel that accepts/transmits data words and memory addresses from many external devices or subchannels, all of which may be in operation at the same time. The system is capable of transmitting up to 571,000 words per second simultaneous with computation (see "Data Multiplexing System").

In summary, considerable input/output flexibility and convenience is afforded the 940 user. For example, the external equipment may include an interlace register which allows entire blocks of data to be entered into or read from memory. Telemetry data may be automatically multiplexed or decommutated, obviating sorting and sequencing within the computer.

COMMUNICATION CHANNEL DESCRIPTION

Up to 30 peripheral devices may be attached to a channel. Each of these devices has a unique, 6-bit unit address by which it is selected for an input/output operation. To select the peripheral device, the program loads the proper unit address into the 6-bit unit address register (UAR) in the channel buffer. This address selects both the device and, if appropriate, the function to be performed. When any nonzero unit address is placed in the UAR, the peripheral unit addressed is said to be "connected" to the channel and it is said to be in the "active" state. When the UAR is loaded with a zero address, or any time that a terminal or initial condition causes the contents of UAR to be zero, the channel is "inactive" and no peripheral unit is "connected" to the channel.

When the channel and the peripheral unit to be used have been connected, the channel must have information pertaining to the location in memory of the data to be transmitted or recieved and pertaining to the number of data words in the transfer.

The number of data words to be in a data transfer is loaded into the word count register (WCR). This 15-bit register contains the data word count during the transmission. This count is decremented and replaced in the WCR for each word transmitted. When the word count is equal to zero, the transmission is complete; in which case, the channel automatically disconnects from the peripheral device and becomes inactive.

The starting memory destination (or source address) for the transmitted data is contained in the memory address register (MAR). The memory address of the location to (or from) which data words are to be transmitted is loaded into the MAR at the same time the word count is loaded. During transmission of the data, the contents of the MAR are incremented after each word just as the contents of the WCR are decremented.

TIME-MULTIPLEXED CHANNEL REGISTERS

In the time-multiplexed channels (see Figure 6), there are two other registers besides UAR, WCR, and MAR just discussed that are important to the programmer; these are the word assembly register (WAR) and the single-character register (SCR). The WAR is a 24-bit buffer that contains the word of data actively being received or transmitted during an input or output operation. During input, 6-bit characters are received into the SCR and assembled one at a time into the WAR; then, the completed word is placed in memory. Depending on the number of characters per word specified, the word placed in memory during input has the form:

One 6-bit character per word

Unpredictable data (0–17)	1st (18–23)

Two 6-bit characters per word

Unpredictable (0–11)	1st (12–17)	2nd (18–23)

Three 6-bit characters per word

Unpredictable (0–5)	1st (6–11)	2nd (12–17)	3rd (18–23)

Four 6-bit characters per word

1st (0–5)	2nd (6–11)	3rd (12–17)	4th (18–23)

When the end of an information record is detected by a channel, the channel automatically disconnects from the device and is then "ready" for another operation. The channel logic is reset, except that the state of the channel error indicator is maintained and the last word of the input is still

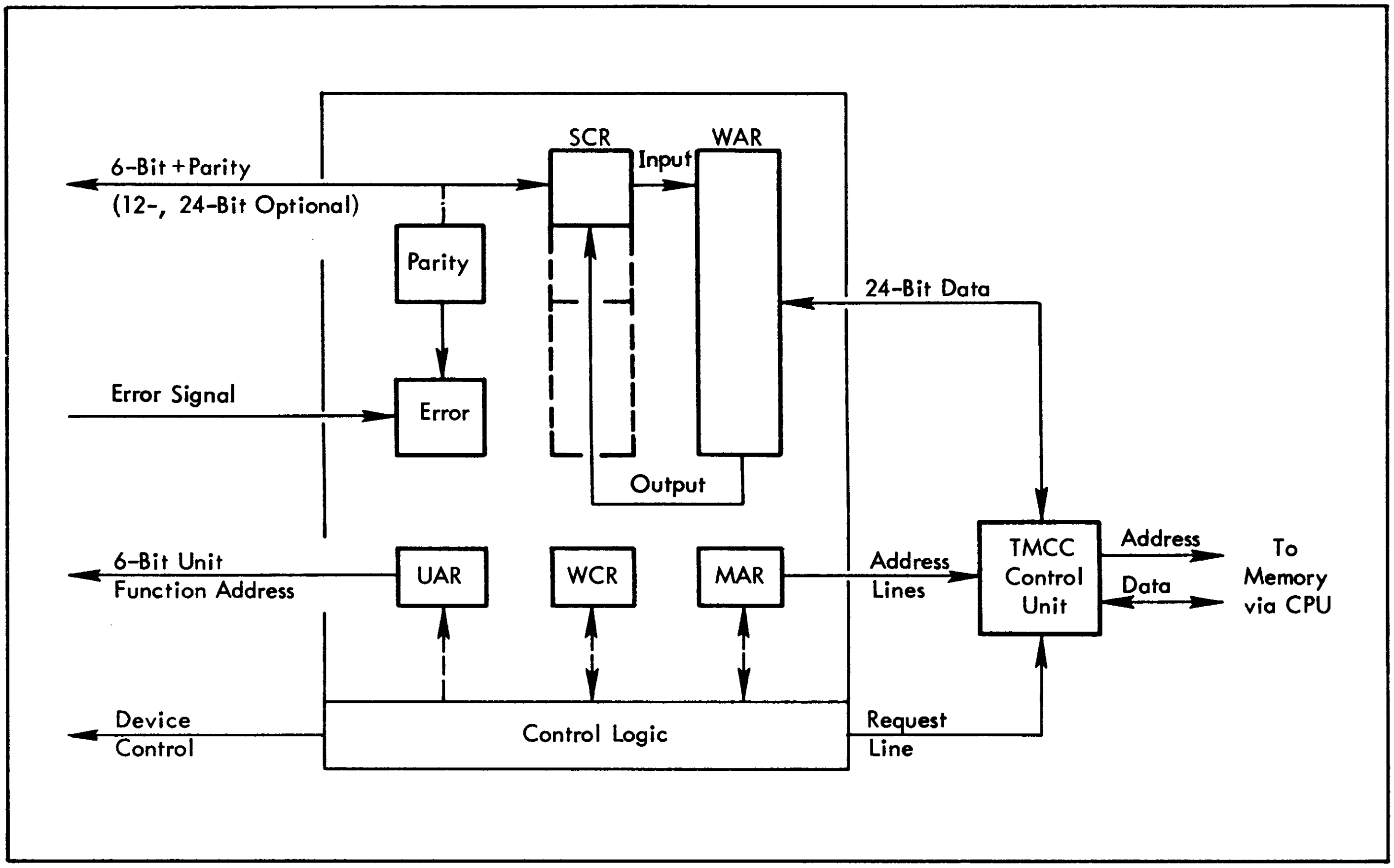


Figure 6. Typical XDS 940 Time-Multiplexed Communication Channel, Block Diagram

in the word assembly register. If the number of characters in the input record was not a multiple of the number of characters assembled into each computer word, then zeros are automatically forced into the least significant positions of the last word. This last word can then be stored in memory by a CHANNEL W INTO MEMORY (WIM) instruction after the channel has disconnected. If the number of characters in the input record was a multiple of the number of characters assembled into each computer word, then the word remaining in the channel buffer is either the last group of characters from the input device, (if they were not previously transferred to memory) or zeros (if the last group of characters had been transferred to memory). In either case, it is safe to issue one such WIM instruction after the channel has disconnected without "hanging up" the computer.

During output, words are brought from memory into the WAR and disassembled into the SCR, one 6-bit character at a time. Depending on the characters/word format specified, the 6-bit characters within the word are output as follows:

Format	Function
One character/word	Output one character from bit positions 0 through 5
Two characters/word	Output two characters from bit positions 0 through 11
Three characters/word	Output three characters from bit positions 0 through 17
Four characters/word	Output four characters from bit positions 0 through 23

As required, the characters are transferred into the single-character register and output. After each character transfer, the word in the WAR is shifted left 6 bits to be ready for the next transfer. Only those characters needed from each word are used; when required, a new word is brought to the WAR for the next character. For special applications, a time-multiplexed channel may be equipped with a 12- or 24-bit single-character register. The external device having a character size greater than 6 bits specifies to the channel what its size is, 12 or 24 bits. Standard 6-bit devices are unaffected by the installation of a wider SCR.

DIRECT ACCESS CHANNEL REGISTERS

In the direct access channels (E through H) the three other registers of importance are the word assembly register (WAR), the input/output register (IOR), and the data chain register (DCR). The WAR is a 24-bit buffer that contains the information actively being transmitted to, or received from, the external device. Information is assembled into, or disassembled from, the WAR in either character or word format; the format is programmer-selectable. In word format, a data word of up to 24 bits is received from a peripheral unit, placed directly into the WAR, and then delivered directly to the IOR. When transmitting in the word format mode, words are delivered directly from the IOR into the WAR and from the WAR to the peripheral unit. When transmitting or receiving words, any size from one bit to 24 bits is acceptable (see Figure 7).

The IOR is a 24-bit buffer between the WAR and memory. The direct access channel control unit places words into the IOR, awaiting their transfer to WAR to be output. During input, the IOR receives words from the WAR and places them into memory under control of the word count and memory address being used in the transmission.

When operating in the character mode, one to four characters are packed into a word. These will normally be the standard 6-bit input/output character size. Characters of less than 6 bits can be handled in character format as defined by a particular installation's need. For character formats that use characters of less than 6 bits, the data transmission is actually in 6-bit character form with zeros filling out the remainder of the 6 bits. When operating in character format mode, the number of characters to be packed into, or unpacked from, each data word can be specified by program control. Under this format, one, two, three, or four characters may be packed into, or unpacked from, all words in a particular data transmission. This is true for all channels.

When receiving 6-bit character data from a peripheral device, the first character of a word is received into bit positions 18 through 23 of the WAR. When the second character is received, the character in bit positions 18 through 23 is shifted into bit positions 12 through 17 and the incoming character is placed into bit positions 18 through 23. The third incoming character causes the characters in bit positions 12 through 23 to be shifted to bit positions 6 through 17 and the incoming character is placed into bit positions 18 through 23. The fourth character causes another 6-bit left shift and then the character is placed in the vacated bit positions 18 through 23. At this point the WAR is completely filled; this information is now copied into the IOR to be placed into the proper memory location. The above procedure would be followed when four characters per data word were specified for the data transmission. If three characters per word had been specified, the WAR would contain three 6-bit characters in bit positions 6 through 23 and zeros in bit positions 0 through 5 when the word is delivered to the IOR. The next incoming character would be accepted as the first of another set of three characters. If two characters per word had been specified, the data word containing two 6-bit characters in bit positions 12 through 23 and zeros in positions 0 through 11 would be delivered to the IOR. If one character per word had been specified, the data word delivered to IOR would contain zeros in bit positions 0 through 17 and one character in positions 18 through 23.

When transmitting data using the character format mode, characters are taken from the most significant end of the

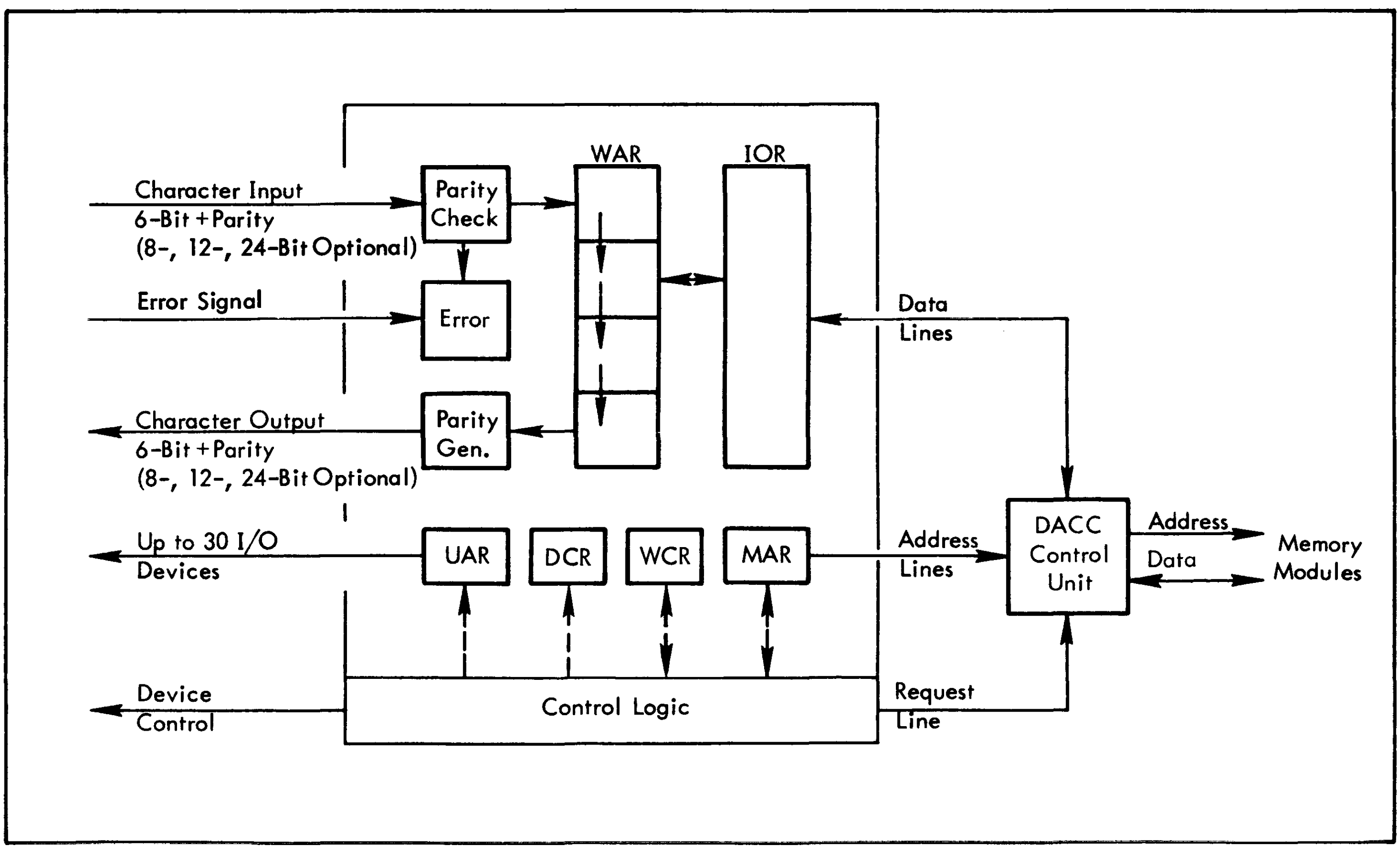


Figure 7. Typical XDS 940 Direct-Access Communication Channel, Block Diagram

WAR. If one character per word is specified, the 6-bit character in bit positions 0 through 5 of the WAR is transmitted to the external device and then another word of information is received from the IOR. If two characters per word are specified, the 6-bit character in positions 0 through 5 of the WAR is transmitted. Then the contents of bit positions 6 through 23 of the WAR are shifted left into positions 0 through 17, the new 6-bit character in positions 0 through 5 is transmitted, and another word is accepted from the IOR to be processed. If three characters are specified, the 6-bit character in bit positions 0 through 5 of the WAR is transmitted. The contents of the WAR are shifted 6 bit positions left and the new contents of bit positions 0 through 5 are transmitted. The contents of the WAR are again left-shifted 6 bits and the third character from positions 0 through 5 is transmitted; then, another word is received from the IOR to be processed. If four characters are specified, the above process is extended to one more 6-bit left shift and the final 6 bits of the word are transmitted before the next word is accepted from the IOR.

The data chain register (DCR) is used to control input/output operations that involve data chaining (i.e., data transfer to/from memory locations in noncontinuous memory blocks). The DCR is a 6-bit register that is loaded with a 1-bit data chain interrupt flag and the 5-bit actual block number of the next memory block associated with the I/O operation. After the DCR has been loaded, the channel control unit monitors the contents of the memory address register (MAR). When the 11 low-order bits of the MAR are all zeros after being incremented, the 5 high-order bits of the MAR are replaced by the contents of the DCR so that the I/O operation continues with a new memory block. If the data chain interrupt flag has been set to 1, the channel transmits a signal to the zero-word-count interrupt level associated with the channel at this time; otherwise, the computer is not notified when the data chaining occurs (see "Data Chaining").

PRIMARY INPUT/OUTPUT INSTRUCTIONS

EOM ENERGIZE OUTPUT M (Privileged)

	02		M	Address

0 2 3 8 9 10 11 12 23

The major instruction for preparing channel W, Y, C, or D and an attached peripheral device to perform a data transmission or other operation in ENERGIZE OUTPUT M (EOM). This instruction operates in four distinct modes: buffer control (mode 0), input/output control (mode 1), internal control (mode 2), and system control (mode 3). In modes 2 and 3, EOM is used in non-communication channel operations such as special systems transmission. The different modes of operation are program selectable by the setting of two bits (10, 11) within the EOM instruction format as follows:

10	11	Mode	Function
0	0	0	Buffer control
0	1	1	Input/output control
1	0	2	Internal control
1	1	3	System control

EOM in the buffer control mode (0) operates essentially as a set-up or preparation facility for data transmissions or other peripheral activities using the channel. The channel to be used, the peripheral unit on that channel, the operation to be performed, and the type of character format to be used are all detailed within the EOM in this mode. The use of BCD or binary transmission, the allowance or not of a leader (as in paper tape), and the direction of operation (as in forward direction for magnetic tape) are all detailed to the channel and its "connected" peripheral unit. Execution of such an EOM connects the specified peripheral unit to the channel buffer, starts the device (if it is in a ready condition), and alerts the channel interlace, if desired.

The EOM in the input/output control mode (1) is used to direct peripheral devices to perform nontransmitting operations such as rewind magnetic tape and upspace the printer. Selection of certain channel operations such as interrupt response and input/output terminal function desired is made with this EOM. It is also used to alert peripheral devices that a PARALLEL INPUT (PIN) or PARALLEL OUTPUT (POT) instruction is to follow. An extension of the word count to 15 bits for the number of words to be transmitted and an extension of the address specification to 16 bits can be given in this EOM.

The EOM in the internal control mode (2) is used to enable and disable the interrupt system. EOM in this mode is also used to prepare the system for the selective arming and disarming of the system interrupt levels.

EOM operating in the system control mode (3) is used to transmit information which is specifically coded for a given installation and system. Address capability is expanded for special system designations.

If an interrupt occurs during the execution of an EOM in any mode, it will not be acknowledged until the execution of the instruction following the EOM is completed.

Affected: determined by address field Timing: 1

EOD ENERGIZE OUTPUT TO DIRECT ACCESS CHANNEL (Privileged)

	06		M	Address

0 2 3 8 9 10 11 12 23

The EOD instruction operates in buffer control (0) and input/output control (1) modes. It refers to channels E, F, G, and H and performs essentially the same functions and operations as an EOM. The internal control (2) and system control (3) modes are available, as special systems require expanded capabilities.

Affected: determined by address field Timing: 1

SKS SKIP IF SIGNAL NOT SET (Privileged)

	40		M	Address

0 2 3 8 9 10 11 12 23

The SKS instruction is the principal instruction for testing the states and responses of a channel and its attached peripheral devices. SKS is a skip class instruction yielding a decisional and transfer capability to the input/output system. This instruction also operates in four distinct modes: special internal test (mode 0), channel and device test (mode 1), internal test (mode 2), and special system test (mode 3). In modes 2 and 3, the SKS is used to test non-channel-oriented functions. These different modes of operation are program selectable by the setting of two bits (10, 11) within the SKS instruction format:

10	11	Mode	Function
0	0	0	Special internal test
0	1	1	Channel and device test
1	0	2	Internal test
1	1	3	Special system test

In the channel and device test mode (1), SKS can be programmed to test a channel for channel active, word count equals zero, channel interrecord condition, and for channel error. This mode also tests peripheral devices directly. These tests include testing indicators in a magnetic tape unit such as beginning of tape, end of tape, file protect ring present, and end-of-file. For example, an indicator within the printer might be addressed by an SKS instruction to determine whether the paper is at the end-of-form.

In the internal test mode (2) SKS tests whether or not the interrupt system is enabled.

In the special internal test (0) and special system test (3) modes, SKS tests signals of special configurations as the specific systems require.

Affected: (P)	Timing:	Modes 0 & 2	Modes 1 & 3
		1 if no skip	2 if no skip
		2 if skip	3 if skip

DIRECT PARALLEL I/O INSTRUCTIONS

Two instructions, PARALLEL OUTPUT (POT) and PARALLEL INPUT (PIN), permit any word in core memory to be presented in parallel at a peripheral connector; or, inversely, permit signals sent to a connector to be stored in any core memory location. The execution of a PIN instruction causes a signal to be sent to the external device involved in the input operation. This signal alerts the device to send its data word as soon as it is operational. When it 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. 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.

POT and PIN can be used effectively with the EOM/EOD instructions to produce high-speed, synchronized, data transfers without the use of a communication channel. Selective input or output from and to a number of external holding devices is accomplished by preceding POT and PIN with a special systems mode EOM or EOD to select the desired device. By preceding the POT or PIN with a special system mode SKS, the ready signal can be tested prior to execution of the parallel transfer instruction and the hang-up of the computer can be avoided. If the ready signal from the external device is used to initiate one of the priority interrupts, 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 is still present when the POT or PIN is executed, no hang-up will occur. These features allow the computer to execute parallel input/output operations at rates up to 70,000 words per second.

During transmissions of this type, no other computation can be performed. Direct access channel transfers can occur simultaneously and at their usual rates, provided that memory modules associated with the channels are different from memory modules associated with the parallel input/output transmissions.

PIN PARALLEL INPUT (Privileged)

U	X	0	33	I	Reference address
0	1	2	3 – 8	9	10 – 23

PIN stores the contents of 24 input lines in the effective location.

Affected: (EL) Timing: 4+wait

POT PARALLEL OUTPUT (Privileged)

U	X	0	13	I	Reference address
0	1	2	3 – 8	9	10 – 23

POT transmits the effective word to 24 output lines.

Affected: Output device Timing: 3+wait

STANDARD EOM/EOD CHANNEL INSTRUCTIONS

Several EOM and EOD instruction configurations have standard uses. The instructions for channel W are given standard assembler-type mnemonics and are set aside as separate instructions.

ALCW ALERT CHANNEL W

The channel is alerted; however, the channel buffer is not affected in any way. The EOM/EOD instructions (and their octal configurations) that are used to alert the various channels are:

Channel	EOM/EOD	Configuration
W	EOM 50000B	0 02 50000
Y	EOM 50100B	0 02 50100
C	EOM 50000B, 2	2 02 50000
D	EOM 50100B, 2	2 02 50100

Channel	EOM/EOD	Configuration
E	EOD 50000B	0 06 50000
F	EOD 50100B	0 06 50100
G	EOD 50000B, 2	2 06 50000
H	EOD 50100B, 2	2 06 50100

DISW DISCONNECT CHANNEL W

The channel is disconnected. Its unit address register is unconditionally set to 00 regardless of whether a device is currently being addressed by the channel. Any device which is connected to the channel is disconnected from the channel. The EOM/EOD instructions (and their octal configurations) that are used to disconnect the various channels are:

Channel	EOM/EOD	Configuration
W	EOM 0	0 02 00000
Y	EOM 100B	0 02 00100
C	EOM 0, 2	2 02 00000
D	EOM 100B, 2	2 02 00100
E	EOD 0	0 06 00000
F	EOD 100B	0 06 00100
G	EOD 0, 2	2 06 00000
H	EOD 100B, 2	2 06 00100

ASCW ALERT TO STORE ADDRESS FROM CHANNEL W

The channel is alerted for a PIN instruction to follow. This instruction does not affect the operation of the channel. See "Direct Parallel Instructions" for a detailed discussion of PIN.

This configuration of EOM/EOD is always used in conjunction with PIN to determine the current completion status of an I/O operation being performed by the selected channel. The two instructions should be written in the order EOM/EOD, PIN. When these two instructions have been executed, the contents of the effective location of the PIN instruction contains the following information:

0–7	8–23
(shaded)	Contents of the channel MAR

The EOM/EOD instructions (and their octal configurations) that are used to alert the various channels to store the contents of their memory address registers are:

Channel	EOM/EOD	Configuration
W	EOM 12000B	0 02 12000
Y	EOM 12100B	0 02 12100
C	EOM 12000B, 2	2 02 12000
D	EOM 12100B, 2	2 02 12100
E	EOD 12000B	0 06 12000
F	EOD 12100B	0 06 12100
G	EOD 12000B, 2	2 06 12000
H	EOD 12100B, 2	2 06 12100

TOPW TERMINATE OUTPUT ON CHANNEL W

This instruction is used to terminate channel output when the last word of a block has been delivered to the channel. After the last character in the channel buffer is delivered to the peripheral device, the channel is disconnected.

This instruction is always used to terminate a noninterlaced channel W output operation. It may be used with all communication channels if the particular function selected is terminal function 11 (IOSP) but no further data output is required (see "Extended Mode Terminal Functions"). The EOM/EOD instructions (and their octal configurations) that are used to terminate output on the various channels are:

Channel	EOM/EOD	Configuration
W	EOM 14000B	0 02 14000
Y	EOM 14100B	0 02 14100
C	EOM 14000B, 2	2 02 14000
D	EOM 14100B, 2	2 02 14100
E	EOD 14000B	0 06 14000
F	EOD 14100B	0 06 14100
G	EOD 14000B, 2	2 06 14000
H	EOD 14100B, 2	2 06 14100

COMPATIBLE/EXTENDED INPUT/OUTPUT MODES

The termination of an I/O operation and the interrupts that may be associated with that termination fall into two classes: compatible and extended. The choice of one of these two modes of input/output operation determines how the system behaves when the termination of an I/O operation occurs.

As mentioned in Section 2, "Interrupt System", interrupts occurring at a single level (e.g., location 31B, etc.) can have different names (e.g., zero word count and end of word). These names reflect the different I/O mode in operation when the interrupt occurs. The differences include the timing of interrupt occurrence relative to the I/O operation and type of interrupt requested.

The compatible mode of operation for channels W, Y, C, and D is directly compatible with the XDS 910/920 mode of I/O (interlaced) operation. The type of interrupts that can be requested are the end-of-word and end-of-transmission interrupts.

The extended mode for all channels expands the I/O capabilities to include the terminal and arming functions discussed below. The types of interrupts that can be requested are the zero-word-count and end-of-record interrupts.

COMPATIBLE MODE TERMINAL FUNCTIONS

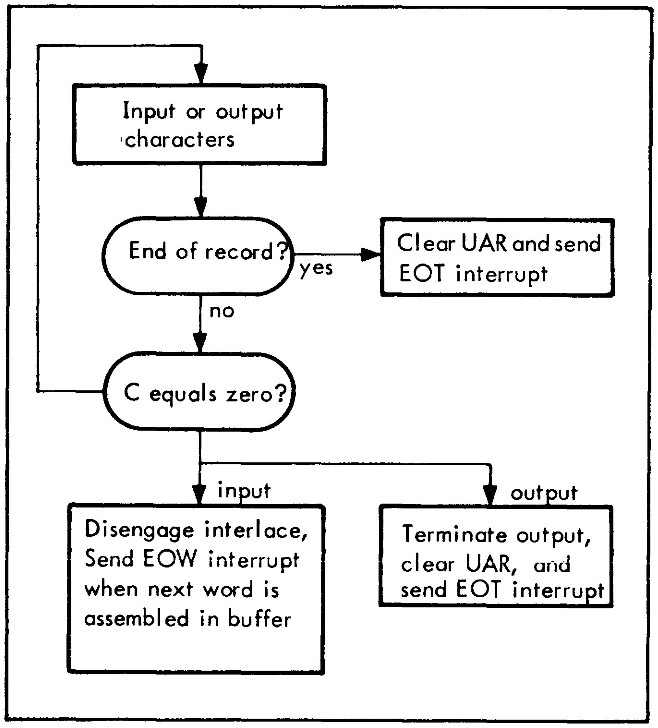
The following description and diagram illustrate the automatic terminal functions of a time-multiplexed communication channel when operating in the compatible interlace mode of data transmission.

Input C Words

If C words are received before the end-of-record detected, the interlace is disengaged and the end-of-word (EOW) interrupt signal is generated when the next word fills the word assembly register. Since the channel continues to accept characters, a character rate error occurs if the record is longer than C+1 words (unless the peripheral device is disconnected from the channel or the remainder of the record is otherwise disposed of). At the end-of-record, the peripheral device is disconnected, the channel becomes inactive, and the channel generates an end-of-transmission (EOT) interrupt regardless of whether or not C words have been read.

Output C Words

When C words are transmitted, the channel is disconnected and the EOT interrupt is generated when the last character is output.



EXTENDED MODE TERMINAL FUNCTIONS

A 2-bit function code in bit positions 15 and 16 of the input/output control mode EOM/EOD (executed after the alert channel EOM/EOD) controls the termination of input/output operations in the extended mode as follows:

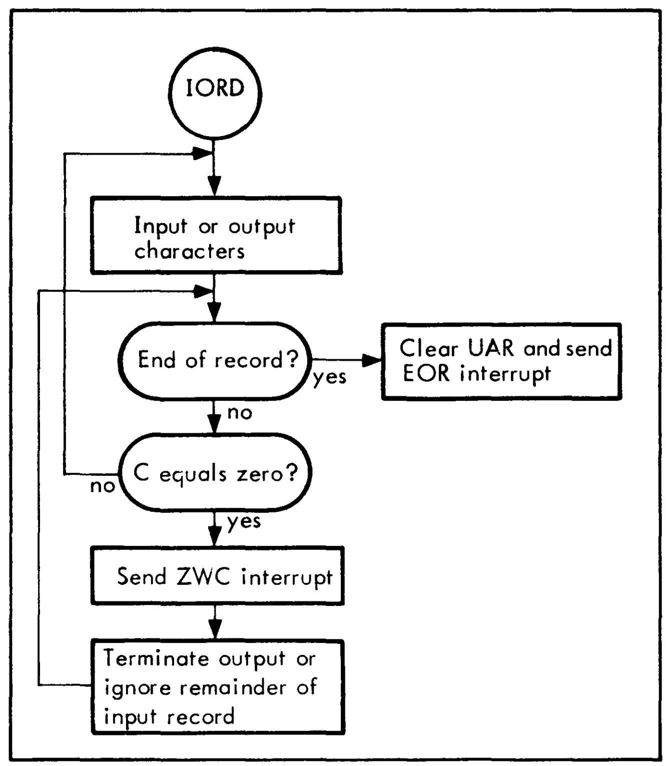
15	16	Terminal Function
0	0	input/output of a record and disconnect (IORD)
0	1	input/output until signal then disconnect (IOSD)
1	0	input/output of a record and proceed (IORP)
1	1	input/output until signal then proceed (IOSP)

These functions are described below with the letter C representing the specified word count of the transmission. Following each of the discussions is a diagram representing the automatic terminal actions of the channel while under control of the specified terminal function in the extended interlace mode of data transmission.

The following table summarizes the terminal functions that should be used with various devices. The IOSP can always be used with any device; however, it causes a disconnect only when an end-of-record signal is received by the channel from a peripheral device during input.

Device	Input	Output
Typewriter	IOSD	IOSD
Paper Tape	IORD, IOSD (IORP can be used but there is no advantage to doing so)	IOSD
Cards	IORD, IOSD (this should not normally be used to disconnect in the middle of a half-read card)	IOSD, IORD
Printer		IOSD, IORD
Magnetic Tape	IORD, IORP	IORD, IORP

Input/Output of a Record and Disconnect



A program should not use the IORD function with devices that do not have end-of-record conditions on output (e.g., devices such as the paper tape punch and typewriter). These devices do terminate output but give the program no indication when they receive the last characters.

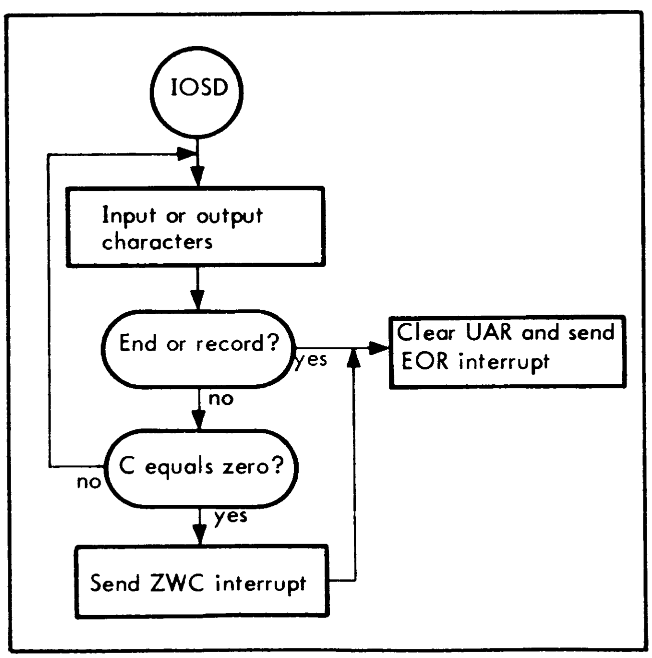
Input C Words. If C words are received before the end-of-record is detected, the rest of the record is ignored. At the end-of-record, the peripheral device is disconnected and the channel becomes inactive.

Output C Words. When C words are transmitted, the device is signaled that the last characters have been transmitted. When the peripheral device has generated the end-of-record and, if necessary, checked the validity of the record, it sends an end-of-record response to the channel, which causes the channel to disconnect and an end-of-record interrupt (if armed).

The line printer generates the end-of-record response when it completes the printing of a line. If the printer encounters any print error or faults, it sends a signal to the channel that sets the channel error indicator; this can occur since the printer has not disconnected from the channel. The IORD function is useful when the program is to print several lines and the program is not otherwise to use the channel between lines. When the printer completes each line, it causes an end-of-record interrupt (assumed to be armed), notifying the program that it can immediately transmit the next paper control instruction and the next line image.

The unbuffered card punch operates similarly. It generates the end-of-record response after punching each row. If any faults occur during the punching of the entire card, the card punch sends a signal to the channel that sets the channel error indicator; this occurs after punching the last row (row 9).

Input/Output until Signal then Disconnect

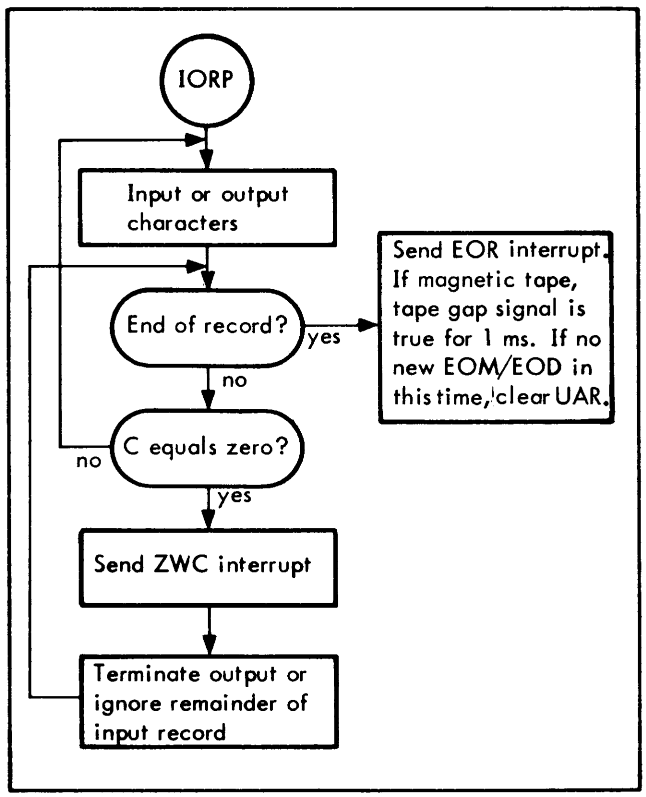


The IOSD function is designed for use on devices that are normally operated on the basis of the word count only. Typewriters and paper-tape devices are of this type, as are the printer and card punch when the user does not wish to stay connected until the operation is complete.

Input C Words. When C words are received, or when the end-of-record is encountered, the device is disconnected and the channel becomes inactive. If the channel disconnects because of zero word count, an end-of-record interrupt (if armed) will be generated in addition to the zero word count interrupt; if both are armed, the zero word count interrupt will occur first.

Output C Words. When C words are transmitted, the channel disconnects the device and becomes inactive when the last character has been transmitted. If an end-of-record signal is received before the count reaches zero, the channel will disconnect immediately.

Input/Output of a Record and Proceed



A program should not use the IORP function with devices that do not generate end-of-record responses upon output termination; such devices are paper tape and typewriter. These devices do terminate output but give the program no indication when they receive the last character.

The IORP should also not be used with the printer and card punch since these devices expect the channel to disconnect after they send an end-of-record signal to the channel.

Input C Words. If the channel receives C words before the peripheral device encounters the end-of-record, the channel ignores the rest of the record (to the end-of-record). When the peripheral device sends the end-of-record signal to the channel, the channel sets its end-of-record indicator; this signal sets the end-of-record interrupt (if armed). The channel does not disconnect. The channel is now in an "interrecord" condition.

When the peripheral is magnetic tape, the tape continues to move when the tape handler encounters the end-of-record. The end-of-record occurs when the tape read-heads encounter tape gap; this also causes a tape signal to "come high". If the program executes a new read-tape or scan-tape EOM/EOD during the intra-gap time (approximately one millisecond, while the tape gap signal is high), the tape remains in motion and proceeds to read or scan the next record. If the program executes no such EOM/EOD before the tape gap signal drops, the channel disconnects and the tape comes to a stop. No additional interrupt occurs. This is the only condition that causes a channel to disconnect automatically.

All other input devices remain connected until the program takes further action. The paper tape reader remains in motion; the program should issue a disconnect channel EOM/EOD if the program is not reading any more tape. To proceed after the end-of-record occurs, the program first reloads the interlace portion of the channel and then executes a buffer control mode EOM/EOD to reinitialize the channel. Otherwise, the channel immediately terminates any attempt to use its interlace portion since the channel is aware that it is still active and in the end-of-record condition. When the program continues from an interrecord condition, the program should use an extended-mode terminal function. An IORP function should not be used to read from or write with devices that do not have end-of-record signals (e.g., the typewriter and paper tape punch).

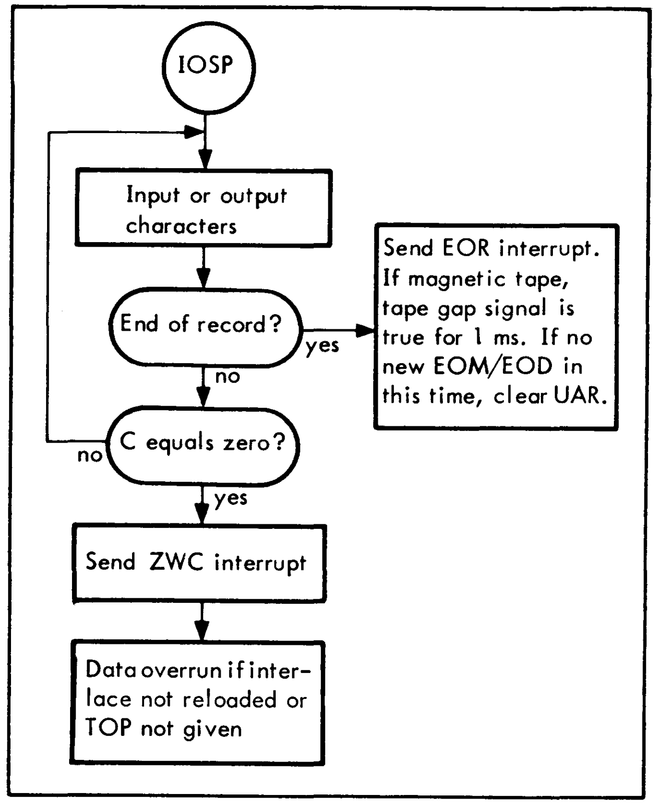
Output C Words. When the channel transmits C words, it sends a signal to the connected peripheral device indicating that the device has the last word. When the peripheral device receives and checks the validity of this last word, it sends an end-of-record response to the channel, which causes the channel to generate an end-of-record interrupt (if armed) and set the interrecord indicator; the channel does not disconnect.

When the peripheral device is magnetic tape, the tape continues to move after it signals end-of-record. As in reading tape, the signal causes the tape gap signal to come high.

If the program executes a new write-tape or erase-tape EOM/EOD during the intra-gap time (approximately one millisecond), the tape remains in motion and proceeds to write or erase a new record. If the program executes no such EOM/EOD before the tape gap signal drops, the channel disconnects and the tape comes to a stop. No interrupt occurs at this time. This is the only condition that causes a channel to disconnect automatically.

To proceed after the end-of-record occurs, the program first reloads the interlace portion of the channel and then executes a buffer control mode EOM/EOD to reinitialize the channel UAR. Otherwise, the channel immediately terminates any attempt to use its interlace portion, since the channel is aware that it is still active and in the end-of-record condition. When the program continues from an interrecord condition, the program should use an extended-mode terminal function.

Input/Output until Signal then Proceed



Input C Words. If the channel receives C words before the peripheral device encounters the end-of-record, the channel generates a zero word count interrupt (if armed). The program should reload the interlace portion of the channel to continue reading the record. Failure to reload the interlace before the peripheral device sends enough characters to overfill the word assembly register causes a character rate error; this sets the channel error indicator.

When the peripheral device encounters the end-of-record, the IOSP function operates identically like the IORP function.

Output C Words. When the channel has transmitted C words, it generates a zero word count interrupt (if armed); the channel does not terminate output. The program should reload the interlace portion of the channel to continue writing in the same record. Failure to reload the interlace before the peripheral device requests the next character causes a character rate error; this sets the channel error indicator.

If the program executes a terminate output EOM/EOD after the channel has counted C down to zero, the channel terminates the output and operates as an IORP function from this point on.

INPUT/OUTPUT CONTROL MODE EOM/EOD

The input/output EOM/EOD selects the I/O operation mode. When the extended mode is selected, this EOM also selects (arms) interrupts that are to be made operational and selects the desired terminal function.

0	02/06	I	0	1	CE	ER	ZC	F	C	A	Hi count
0 2	3 8	9	10	11	12	13	14	15	16	17 18	19 23

Bit Position	Function
0, 1, 2	Bit positions 0, 1, and 2 are ignored.
3-8	Bit positions 3 through 8 contain the instruction code for EOM(02) or EOD(06).
9(I)	Bit position 9 alerts the interlace. If the interlace has been alerted by a previous alert channel EOM, this bit is ignored.
10, 11	Bit positions 10 and 11 contain the EOM indicator for the input/output control mode (mode 1).
12(CE)	Bit position 12 selects the mode of I/O operation. A 0 specifies the compatible mode. The operation of bits 13, 14, 15, and 16 are disallowed. Channels W, Y, C, and D only can operate in this mode; if interrupts are required, the user enables the interrupt system (EIR), thus enabling and arming the end-of-word and end-of-transmission interrupts. A 1 specifies the extended mode. All channels can operate in this mode. This allows the use of bits 13, 14, 15, and 16. If interrupts are required, the user arms the associated ones by placing 1-bits in bit 13 and/or 14. The terminal function to be used is selected via bits 15 and 16. Note that a 1 in bit position 13 and/or 14 does the following: 1. Arms that interrupt during this complete I/O operation; disconnecting this channel disarms the interrupt. 2. Once armed by bits 13 and/or 14, the interrupt can be enabled by the enable/disable feature of the interrupt system. If a channel generates an extended mode I/O interrupt while the system is disabled, the designated interrupt level goes to the waiting state. When the program again enables the interrupt system, the interrupt goes to the active state when its priority allows. Direct access communication channels operate only in the extended interlace mode; therefore, a DACC does not examine bit 12, but assumes it to be a 1. Note that with the complete omission of this EOD, a DACC operates in the terminal function 00 (IORD) mode.
13(ER)	Bit position 13 controls the arming of the end-of-record interrupt. A 1 arms the interrupt; a 0 disarms the interrupt.
14(ZC)	Bit position 14 controls the arming of the zero word count interrupt. A 1 arms the interrupt; a 0 disarms the interrupt.
15, 16(FC)	Bit positions 15 and 16 specify the terminal condition function to be performed with the transmission. These are defined in "Extended Mode Terminal Functions".
17, 18(A)	Bit positions 17 and 18 contain the high-order memory address bits.
19-23	Bit positions 19 through 23 contain the 5 most significant bits of the 15-bit word count. These positions specify a word count greater than 1023.

BUFFER CONTROL MODE EOM/EOD

The communication channel EOM/EOD instruction is used in the buffer control mode to specify the channel to be used in the I/O operation, the direction in which the designated I/O device is to operate, the information format, and the number of characters per word. This EOM/EOD also "connects" the channel to the designated I/O device. For an I/O operation, the term "connect" has the following meaning:

a. the address code of the designated I/O device is loaded into the channel's unit address register (UAR)

b. the device is started

c. characters are transmitted from the channel to the device or from the device to the channel

The detailed instruction format is:

0	B1	0	02/06	I	0	0	F/R	L/N	D/B	C/W	B2	Unit
0	1	2	3–8	9	10	11	12	13	14	15 16	17	18–23

Bit Position	Function
0, 2	Bit positions 0 and 2 are not used.
1, 17 (B1, B2)	Bit positions 1 and 17 specify the channel to be activated. For time-multiplexed communication channels (EOM), channel W is numbered 00, channel Y is 01, channel C is 10, and channel D is 11. For direct access communication channels (EOD), channel E is numbered 00, channel F is 01, channel G is 10, and channel H is 11.
3-8	Bit positions 3 through 8 contain the instruction code for EOM(02) or EOD(06).
9(I)	Bit position 9 alerts the channel interlace. A 0-bit specifies noninterlace operation. A 1-bit specifies interlace operation. If a previous EOM/EOD has alerted the interlace, this bit is ignored.
10, 11	Bit positions 10 and 11 contain the EOM mode indicator for the buffer control mode (mode 0).
12(F/R)	Bit position 12 specifies the direction in which the peripheral device will operate. A 0 specifies the forward direction. A 1 specifies the reverse direction. The reverse direction should only be specified for those devices that can

Bit Position	Function
	operate in reverse; if the device cannot operate in reverse, an unpredictable operation occurs.
13(L/N)	Bit position 13 specifies whether the paper tape punch should be started with a leader. A 0 specifies a start with leader and a 1 specifies a start without leader. The start with leader specification should only be given for a paper tape punch; if any other device is instructed to start with leader, an unpredicatable operation occurs.
14(D/B)	Bit position 14 specifies the mode of character format: a 0 specifies BCD format, and a 1 specifies binary format.
15, 16(C/W)	Bit positions 15 and 16 specify the number of characters to be assembled into, or disassembled from, each transmitted word. One character per word is specified by 00, two by 01, three by 10, and four by 11.
18-23 (Unit)	Bit positions 18 through 23 specify the unit and the function to be performed with that unit (see Table 4.)

CHANNEL AND DEVICE SKS

The SKIP IF SIGNAL NOT SET (SKS) is used in the channel and device test mode (mode 1) as described below. The SKS tests the indicators in a channel as well as devices attached to it. To test the channel, unit address 00 is used. Instruction format is:

0	C2	0	40	C1	0	1	R	C	E	I	0	C3	00
0	1	2	3–8	9	10	11	12	13	14	15	16	17	18–23

Bit Position	Function
0, 2, 16	Bit positions 0, 2, and 16 are ignored.
9, 1, 17 (C1, C2, C3)	Bit positions 9, 1, and 17 are used as an octal digit to specify the channel to be tested. Channel W is 0, channel Y is 1, and so on, channel H being 7.
3-8	Bit positions 3 through 8 contain the operation code for SKS.
10, 11	Bit positions 10 and 11 contain the mode selection for mode 1.

Table 4. Unit Address Codes

00	Disconnect	40	Not Used
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 Reader Input No. 1	44	Paper Tape Punch Output No. 1
05	Paper Tape Reader 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	Rapid-Access Data File Input No. 1	66	Rapid-Access Data File Output No. 1
27	Rapid-Access Data File Input No. 2	67	Rapid-Access Data 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

Bit Position	Function
12(R)	Test for ready; a 1 selects the test.
13(C)	Test for word count equal to zero; a 1 selects the test.
14(E)	Test for error indicator reset; a 1 selects the test.
15(I)	Test for interrecord condition; a 1 selects the test.
18-23	Bit positions 18 through 23 are zero to specify a data channel test.

Note: If more than one test is selected (i.e., if bit positions 12 through 15 contain more than one 1), the results of the test are unpredictable.

Affected: (P) Timing: 2 if no skip
3 is skip

STANDARD CHANNEL TEST INSTRUCTIONS

Several SKS function configurations have standard uses. These are given standard assembler-type mnemonics and are always used as shown.

CATW CHANNEL W ACTIVE TEST
(Skip if channel W not active)

If the channel is ready to accept a new input/output instruction, the next instruction in sequence is skipped and the following instruction is executed. If the channel is active, busy servicing a command, or in the process of disconnecting a peripheral unit, the next instruction in sequence is executed. The SKS instructions (and their octal configurations) for testing channel active are:

Channel	SKS	Configuration
W	SKS 14000B	0 40 14000
Y	SKS 14100B	0 40 14100
C	SKS 14000B, 2	2 40 14000
D	SKS 14100B, 2	2 40 14100
E	SKS 54000B	0 40 54000
F	SKS 54100B	0 40 54100
G	SKS 54000B, 2	2 40 54000
H	SKS 54100B, 2	2 40 54100

CETW CHANNEL W ERROR TEST
(Skip if no error on channel W)

The error indicator in the channel is tested for being in the set condition. If the error indicator has not been set, the next instruction in sequence is skipped and the following instruction is executed. If the error indicator has been set, the next instruction in sequence is executed. The SKS instructions (and their octal configurations) for testing channel error are:

Channel	SKS	Configuration
W	SKS 11000B	0 40 11000
Y	SKS 11100B	0 40 11100
C	SKS 11000B, 2	2 40 11000
D	SKS 11100B, 2	2 40 11100
E	SKS 51000B	0 40 51000
F	SKS 51100B	0 40 51100
G	SKS 51000B, 2	2 40 51000
H	SKS 51100B, 2	2 40 51100

CZTW CHANNEL W ZERO COUNT TEST
(Skip if channel W word count is zero)

A test is made on whether the contents of the word count register in the channel have been reduced to zero. If the contents of the word count register are zero, the next instruction in sequence is skipped and the following instruction is executed. If the contents of the word count register are nonzero, the next instruction in sequence is executed. The SKS instructions (and their octal configurations) for testing channel word count are:

Channel	SKS	Configuration
W	SKS 12000B	0 40 12000
Y	SKS 12100B	0 40 12100
C	SKS 12000B, 2	2 40 12000
D	SKS 12100B, 2	2 40 12100
E	SKS 52000B	0 40 52000
F	SKS 52100B	0 40 52100
G	SKS 52000B, 2	2 40 52000
H	SKS 52100B, 2	2 40 52100

CITW CHANNEL W INTERRECORD TEST
(Skip if channel W interrecord condition present)

A test is made to determine if the channel is in the interrecord condition. This condition is present or true only after an IOSP or IORP instruction has encountered an end-of-record on input or output, and remains true as long as the channel is active (i.e., until the device is disconnected), or until a continuing EOM/EOD followed by the loading of a new instruction occurs. If the interrecord condition is true, the next instruction in sequence is skipped and the following instruction is executed. If the interrecord condition is not true, the next instruction in sequence is executed. The SKS instructions (and their octal configurations) for testing channel interrecord condition are:

Channel	SKS	Configuration
W	SKS 10400B	0 40 10400
Y	SKS 10500B	0 40 10500
C	SKS 10400B, 2	2 40 10400
D	SKS 10500B, 2	2 40 10500
E	SKS 50400B	0 40 50400
F	SKS 50500B	0 40 50500
G	SKS 50400B, 2	2 40 50400
H	SKS 50500B, 2	2 40 50500

DEVICE TEST INSTRUCTIONS

The SKIP IF SIGNAL NOT SET (SKS) below is used in the channel and device test mode (mode 1) to test the condition of the peripheral devices in the system directly. The individual test instructions are described in Section 6.

0	C2	0	40	C1	0	1	Test	C3	Unit
0	1	2	3–8	9	10	11	12–16	17	18–23

Bit Position	Function
0, 2	Bit positions 0 and 2 are ignored.
9, 1, 17 (C1, C2, C3)	Bit positions 9, 1, and 17 are used as an octal digit to specify the channel. Channel W is 0, channel Y is 1, and so on.
3-8	Bit positions 3 through 8 contain the operation code for SKS.
10, 11	Bit positions 10 and 11 contain the selection for mode 1.
12-16	Bit positions 12 through 16 select the particular test and are device dependent.
18-23	Bit positions 18-23 contain the unit address of the I/O device to be tested (see Table 4).

PROGRAMMING THE INTERLACE REGISTER

Programming a block transmission of data using the full facility of the 940 input/output system includes an alert channel and buffer setup EOM/EOD, an interlace setup EOM/EOD (optional), and a POT instruction.

The first EOM/EOD in an interlaced input/output sequence is always a buffer control mode EOM/EOD that alerts the appropriate channel interlace, connects the channel to the peripheral device and starts the data transfer. However, the buffer setup EOM/EOD may be executed at a later time, as in the case of waiting for a device to become ready after the channel interlace has been set up.

The second EOM/EOD (optional) is an input/output control EOM/EOD that specifies the interrupts and the terminal function to be used. It also can specify two high-order address bits and five high-order word count bits; these expand the starting address to a maximum of 65,535 and the word count to a maximum of 32,767.

When using time-multiplexed channels W, Y, C, or D where the input/output terminal functions, interrupts, and additional count or address bits are not desired, the second EOM is not needed to set up the channel; the end-of-record interrupt for the selected channel occurs if the interrupt system is enabled. If this second EOM is not present, the time-multiplexed channels automatically operate in the compatible mode. When using terminal function control, the eight channels are programmed in the same way.

The second EOM/EOD (if present) is followed immediately by a PARALLEL OUTPUT (POT) instruction that transmits a data word to the interlace registers of the selected channel. The data word transmitted to the interlace register by the POT instruction has the following format:

Word count	Starting address
0 — 9	10 — 23

The word count is right-justified in bit positions 0 through 9 of the word, providing for input/output of up to 1023 words. The starting address of the input/output operation is right-justified in bit positions 10 through 23 of the word.

EXTENDED MODE EXAMPLE

A sample sequence from a magnetic tape read operations is given below.

Location	Instruction	Comments
⋮		
1000B	EOM 42611B	This buffer control mode EOM specifies forward direction of tape motion with no leader and BCD character format, selects the four characters per word assembly mode, connects channel W to tape unit number 1, and starts the data input operation. This EOM also alerts the channel W interlace to expect a second EOM instruction or a POT instruction to follow immediately. No other channel is affected.
1001B	EOM 15001B	This I/O control mode EOM, selects the extended channel interrupt mode, disarms the end-of-record interrupt, arms the zero word count interrupt, selects terminal function 00 (IORD) and specifies high-order word count.
1002B	POT 1020B	This POT transmits to the selected channel the contents of location 1020B. The location contains the word count and the starting location for data input.
⋮		
1020	00313500B	This location contains the low order 10 bits of the word count and the low order 14 bits of the starting address.

The channel assembles the starting address from bits 17 and 18 of the second EOM and from the word transmitted by the POT. In this sample, the starting address for the read operation is 13500B. The word count is assembled from bits 19-23 of the same EOM and from the word transmitted by the POT. In this sample, the word count is 2006B. This is assembled as follows. Bits 19 through 23 of the EOM are 000 01; bits 0 through 9 of the transmitted word are 0 000 000 110. Assembling these bits into one 15-bit count, 000010 000 000 110, the word count becomes 2006B.

Thus, the instruction sequence reads one magnetic tape record of 2006B words into memory starting at location 13500B. When the word count equals zero during the transmission, an interrupt signal is sent to the channel W zero-word-count interrupt level (31B). Any further information is ignored; and, when the tape reaches the end-of-record, it is stopped and disconnected, the channel becomes inactive and the end-of-record interrupt signal is sent to the channel W end-of-record interrupt level (33B).

COMPATIBLE MODE EXAMPLE

In the compatible mode of channel operation, the second EOM may be omitted. This mode allows a word count of up to 1023 (1777B) words and starting addresses up to 16,383 (37777B). The end-of-word and end-of-transmission interrupts are used when interrupts are desired. They can be armed and enabled or disarmed and disabled by the enable/disable instructions. Since the extended mode input/output functions that are specified in the second EOM cannot be used, the latter two interrupts are used along with SKS instructions to determine the terminal conditions of input/output transmissions. This I/O mode operates only for channels W, Y, C, and D.

A sample line print sequence programmed in this mode follows:

Location	Instruction	Comments
⋮		
1000B	EOM 42660B	This buffer control mode EOM alerts the channel W interlace, specifies BCD format, selects the four characters per word assembly mode, connects channel W to line printer number 1, and starts the data output operation.
1001B	POT 1030B	This POT transmits to the channel the contents of location 1030B.
⋮		
1030B	02042000B	This location contains the word count (41B) and the starting address (2000B) for output.

Since the input/output facility is less comprehensive in this mode, the user must be aware of the terminal conditions that will occur. For output, the mode is equivalent to function 00; that is when C words have been transmitted, the output is terminated, and when the last character has been sent, the device is disconnected.

If the interrupt system is enabled, an end-of-record interrupt to location 33B occurs when the device disconnects. No interrupt occurs on level 31B. See the "Terminal Functions" discussion for details.

For input, this mode is equivalent to functions 00 (IORD) and 01 (IOSD) if the end-of-record is encountered before the word count is reduced to zero. If the word count is reduced to zero before the end-of-record is encountered, the interlace portion of the channel disengages all its control of the channel buffer. The buffer continues to assemble characters until a word is completed. If the interrupt system is enabled, the buffer then generates an end-of-word interrupt on level 31B. The program has approximately 1.5 character times to reload the interlace if reading is to continue; otherwise, a character rate error occurs. On channel W, the contents of the buffer can be stored with the WIM instruction. This mode of channel operation should generally not be used for input, unless the length of the input record is fixed and known.

DATA CHAINING

The direct access communication channel data chain register (DCR) contains a 1-bit data chain interrupt flag and a 5-bit memory block number. The DCR is loaded with an EOD-POT sequence in much the same manner as the interlace registers are loaded. In order to perform a data chaining operation, the DCR is first alerted with an input/output control mode EOD. The octal configurations of EOD required for the various channels are:

Channel	Instruction	Configuration
E	EOD 11000B	0 06 11000
F	EOD 11100B	0 06 11100
G	EOD 11000B, 2	2 06 11000
H	EOD 11100B, 2	2 06 11100

The alert to load DCR is followed immediately by a POT instruction that transmits a data word to the selected channel. The data word is assumed to be in the following format:

000000	ZC	Block
0 – 17	18	19 – 23

Bit positions 0 through 17 are ignored.

Bit position 18 contains the data chain interrupt bit, which is used as an interrupt flag. If the interrupt flag is a 1, the channel transmits an interrupt signal to the zero word count interrupt level when data chaining occurs; this signal is not controlled by the arming bit in the input/output mode EOD that sets up the interlace control functions. If the interrupt flag is a 0, the program is not interrupted when data chaining occurs.

Bit positions 19 through 23 contain the actual block number for the next memory block to be used in the data transfer after the interlace operation reaches the highest-numbered memory location in the memory block selected by the EOD-POT sequence that sets up the channel interlace.

After the DCR is loaded by the EOD-POT sequence, the channel is set up for data transfer (see "Programming the Interlace Register"). As the data transfer progresses, the carry from the eleventh to the twelfth low-order bit position of the memory address register is monitored. The occurrence of this carry indicates the transition across a block boundary. Before the next word is processed, the block number in the

DCR is transferred to the 5 high-order bit positions of the memory address register, the data chain control is reset, and, if the data chain interrupt flag is set to 1, the zero word count interrupt level is activated (if the interrupt system is enabled). This action causes further data transfers to take place with a new block of memory, and makes the DCR available for further loading. The zero word count interrupt may be used to notify the computer of this fact. The data chain EOD-POT sequence can then be repeated to set up the DCR for the next block to which chaining is to occur. This action can take place at any time during the transfer of the next 2048 words. All other operations of the channel are carried out as normal including the zero word count interrupt when the interlace count truly goes to zero (this action occurs independent of the setting of the data chain interrupt flag; i.e., the data chain interrupt flag is only applicable to the boundary-crossing use of the zero word count interrupt). If no data chaining is called for, the channel operates in a totally normal fashion.

The DCR and its associated control flip-flops are reset at the completion of a data transfer operation and by pressing the START switch on the control console. Thus, if a data transfer operation terminates prior to the crossing of a block boundary, the DCR and its control flip-flops will be reset in spite of the fact that they were not used. The zero word count interrupt level provides two types of interrupts under the above described mode of operation. These can be distinguished in the program through the use of the channel zero count test instruction.

CHANNEL MEMORY ACCESS PRIORITY

During each memory cycle the control unit interrogates each channel to determine if it needs access to memory. If only one channel requires memory access, the channel is allowed to proceed immediately. If more than one channel requires memory access, the one that is allowed to proceed is determined on the basis of a fixed and a variable priority. The fixed priority is in the order (highest to lowest): direct access channel, time-multiplexed channel, and central processor. Time-multiplexed channels have fixed priority in the order (highest to lowers): D, C, Y, W.

Direct access channels have variable priority that is normally determined by a comparison of the word assembly register in each channel. The channel whose word assembly register has the fewest number of characters remaining to be filled is selected for memory access. For example, if the word assembly register in channel E has one character position unfilled and the word assembly register in channel F has three character positions unfilled, channel E is selected. Thus, each channel increases its priority level as each character is read into the word assembly register. If the contents of the register in two or more channels are equal in characters to be filled and no other channel in the set has fewer characters to be filled, priority is determined in sequence, with channel H having top priority. A direct access channel has priority over the central processor only if a data overrun is imminent.

Note that the number of characters to be placed in the word assembly register at any time is dependent on the characters per word count specified for the transmission. Assume, for example, that in channel E the character count is three characters per word and in channel F the character count is four characters per word. If both channel F and channel E need access to memory simultaneously, and if both have two characters filled in their respective word assembly registers, then channel E gets first memory access since it has only one character place to be filled.

SINGLE-WORD DATA TRANSFER

Channels W and Y can be programmed as single-word input/output buffers. Data transfer is performed under direct program control or with the aid of the interrupt system. Interlace is not used with these instructions.

MIW MEMORY INTO CHANNEL W (Privileged)

U	X	0	12	I	Reference address
0	1	2	3 – 8	9	10 – 23

MIW transfers the contents of the effective word into the channel W word buffer. If necessary, the central processor hangs up until the buffer is empty and ready to accept the data word.

The W buffer must be connected to the desired peripheral device by a previous buffer control mode EOM instruction that selects the channel, the unit address, and all appropriate control functions.

Affected: Channel W Timing: 2 + wait

WIM CHANNEL W INTO MEMORY (Privileged)

U	X	0	32	I	Reference address
0	1	2	3 – 8	9	10 – 23

WIM transfers contents of the channel W word buffer into the effective location. If necessary, the central processor hangs up until the buffer is full and ready to deliver the data word.

Affected: (EL) Timing: 3 + wait

MIY MEMORY INTO CHANNEL Y (Privileged)

U	X	0	10	I	Reference address
0	1	2	3 – 8	9	10 – 23

MIY transfers the effective word into the channel Y word buffer. If necessary, the central processor hangs up until the buffer is empty and ready to accept the data word.

Affected: Channel Y Timing: 2 + wait

YIM CHANNEL Y INTO MEMORY (Privileged)

U	X	0	30	I	Reference address
0	1	2	3 – 8	9	10 – 23

YIM transfers the contents of the channel Y word buffer into the effective location. If necessary, the central processor hangs up until the buffer is full and ready to deliver the data word.

Affected: (EL) Timing: 3 + wait

SINGLE-WORD OPERATIONS

The single-word buffer operation of channel W (or Y) is used in two ways. Data word transfers between channel W and memory can be performed under direct program control. The buffer control EOM and the WIM or MIW are given in sequence and the computer hangs up until the buffer is ready to perform the transfer. This delay is usually due to buffer tie-up while the buffer is actively transferring the previously requested data word.

The tie-up of the central processor can be eliminated if the interrupt system is used with WIM and MIW. Using the interrupt system allows the program to start the device to be used in the transfer, to enable the interrupt, and to continue processing in the main program. When the buffer is ready to receive from, or transfer to, memory, the end-of-word interrupt level (31B) notifies the program that the buffer is ready. A service routine is entered via a BRANCH AND MARK PLACE (BRM) instruction in location 31B. This routine contains the MIW or WIM, which can be executed immediately without computer tie-up.

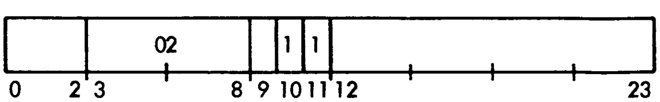
During single-word operations, the channel error indicator is set if a parity error or data overrun is detected. Data overrun occurs when characters enter the buffer: during input, before the previous word is removed; during output, if characters are needed for output before a word is supplied to the buffer. The transmission is not terminated upon detection of any of these errors.

An end-of-record termination can be detected using the interrupt system. During output, use of the TERMINATE OUTPUT (TOP) after the final MIW or WIM causes an end-of-transmission interrupt signal to be sent to level 33B when that final data word has been processed by the buffer (this interrupt signal inhibits the end-of-word interrupt). During input, the end-of-transmission interrupt is sent to location 33B when the end-of-record is detected. During input from devices that do not generate an end-of-record signal, the transmission is terminated using a DISW instruction to disconnect the channel. No end-of-transmission interrupt is received when terminating in this way.

SINGLE-BIT INPUT/OUTPUT

Operating in the system mode (mode 3), the two instructions, ENERGIZE OUTPUT M (EOM) and SKIP IF SIGNAL NOT SET (SKS), are used to provide single-bit input/output transmissions.

Execution of an EOM operating in the system mode causes a 2-microsecond signal to be transmitted to one of a possible 65,536 signal destinations. The system mode EOM format is:



Bit positions 0, 1, and 2 are additional special system address bits.

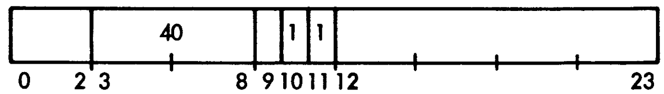
Bit positions 3 through 8 contain the EOM operation code.

Bit position 9 is reserved for use with the Data Multiplex option.

Bit positions 12 through 23 contain the 12-bit address field which specifies the special system destinations.

Affected: Special System Device　　Timing: 1

The SKS system test format is as follows with each corresponding bit-set being identical to the system EOM format:



Execution of an SKS in the system test mode causes a 16-bit address to be presented to the collection of special system devices. If the external device whose address is presented is supplying a set signal to the central processor, the next instruction in sequence from the SKS is executed. If no signal is set, the next instruction in sequence is skipped and the following instruction is executed.

Affected: (P)　　Timing: 2 if no skip
3 if skip

DATA MULTIPLEXING SYSTEM

The standard I/O systems provided with the XDS 940 Computer provide for operation with all standard XDS 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 source of data and for systems which may have very high data rates.

The XDS 940 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 PIN/POT operations use the first path. All time-multiplexed communication channels also use this path. In addition to this path, which is primarily under the control of the main frame, there is a second path that is completely under the control of the units attached to it. This path is made available with the installation of the multiple access to memory feature.

MULTIPLE ACCESS TO MEMORY

The multiple access to memory feature provides the necessary modules on both main frame and memory to permit memories to be accessed via the second path. A word can be transferred over the path in either direction in one cycle. If the computer is equipped with two or more memories and the main frame is communicating with one memory while some other device is using the second path to another memory, then there is no interference with computation. If the second path has priority, the program loses one cycle while the second path transmits.

The multiple access to memory feature is required for the attachment of direct access communication channels (DACC), data multiplex channels (DMC), or memory interface connections (MIC). These devices all incorporate a priority scheme for determining the assignment of the second path. Only four DACCs can be attached to one computer system; memory interface connections, and data multiplex channels, however, are unlimited in number.

DATA MULTIPLEXING BASIC ELEMENTS

A data multiplexing system consists of two basic elements:

1. The data multiplex channel (DMC) for communicating with several data sources/destinations and for synchronizing I/O operations with memory, MICs, DACCs, and other DMCs.
2. One or more data subchannels (DSC) for interfacing between peripheral devices and systems and the DMC.

Data Multiplex Channel (DMC)

The data multiplex channel is the basic unit for the data multiplexing system. It connects to the second path to memory via the multiple access to memory (MAM) feature. A DMC consists of 24-bit register and control logic. All addresses and data are transmitted between the DMC and subchannels via a bus system. The data and address are connected to memory via the MAM only when a transfer is to be made. All program control required for a given I/O operation operates directly on the individual subchannel, not the DMC.

The DMC is equipped with an internal interlace feature. This feature allows a subchannel to specify the address of a word in memory where the data address and count are to be found. When operating with internal interlace, the subchannel supplies the address of its interlace word instead of the actual data address. The DMC reads out the interlace word, increments the address portion, decrements the count, restores the word and then accepts the data from or transmits the data to the subchannel. Transmissions using internal interlace require 3 cycles per word. The DMC also supplies a signal to the subchannel if the decremented count is zero.

The format of the internal interlace word is:

Word count	Data address
0 – 7	8 – 23

The 8-bit word count allows for block lengths of 1 to 256 words (an initial value of 0 is treated as 256).

The DMC also provides for automatic memory incrementing. The counting capability of the DMC register is such that the entire 24-bit register or either the upper 12 bits or the lower 12 bits may 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. The DMC reads out the word, increments it, and then restores it. If the word was zero after the incrementing, the DMC signals the subchannel, which may then interrupt the program. The maximum incrementing rate is 1 count every 2 cycles. Parity generation and detection are available.

Data Subchannels

There are a number of subchannels that can be attached to the DMC. Subchannels can control and generate program interrupts but do not include the interrupt levels themselves. The signals must be routed to optional interrupt levels if the interrupt features are to be used.

The subchannels use a priority scheme to determine which one may transmit to the DMC at any given time. This is similar to the scheme used by the MICs, DMCs, and in transmitting to memory. Up to 128 DSCs can be connected to a DMC. A DSC can use the internal interlace feature of the DMC to control its transmission or it can be equipped with an external interlace (EIN).

A DSC using internal interlace has two words assigned to it. These two words are assumed to be in adjacent even/odd-numbered locations and are fixed for a given subchannel. The program can select either the even- or the odd-numbered location. If the even-numbered location is selected, the subchannel will automatically switch to the odd-numbered location when the count field of the even word is reduced to zero. The program can also select whether or not the subchannel will switch back to the even word when the count field of the odd word is zero. The subchannel will generate an interrupt signal when the count field of either word reaches zero. Transmission termination occurs when the odd word's count equals zero if the subchannel does not switch back to the even word.

The two-word 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 to the other, the program is relieved of the necessity for making real-time responses to the zero count condition. Using first the even then the odd interlace word allows maximum word count of 512 for a pair of interlace 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. It checks and generates the parity of characters to enable it to couple with standard XDS peripherals. The DSC-I has a unit address register.

The subchannel can 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 count is zero and then terminates if interlace cycling is not requested. The output can 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, can always be terminated due to an external end signal. The program can also specify if the DSC is to terminate and disconnect on zero count or disconnect 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.

Word Subchannel (DSC-II). The DSC-II is a general-purpose subchannel designed to allow communication with word-oriented input/output units such as analog-digital and digital-analog converters. It contains no storage for data. 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 that of the DSC-I. The

DSC-II also contains control logic to facilitate memory increment operations in conjunction with the DMC.

EXTERNAL INTERLACE

The external interlace (EIN) 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 transmissions of the DSC at a maximum rate of 1 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 operation performed for internal interlace, except that only two memory cycles are used for interlace word processing. The first is to access the interlace word initially; the second is to restore the interlace word when the count reaches zero.

PROGRAM CONTROL OF DATA SUBCHANNELS

Transmission of data between a DSC and computer memory is controlled by two 24-bit interlace control words unique to the DSC and wired into fixed, adjacent locations in memory. During a transmission the DMC/DSC uses the two interlace control words for determination of transmission address and record length.

The DSCs are numbered from 0 to 0376 in even octal numbers; this permits a maximum of 128 subchannels. The memory locations of the interlace control word pairs associated with the DSCs are numbered X0000, X0001 for DSC-0, X0002, X0003 for DSC-2, . . . , X0376, X0377 for DSC-376. DSC-I numbering need not be contiguous, but DSC-II's are configured one or two in a module and are numbered with adjacent numbers. If a system contains multiple DSC-II modules (each with 1 or 2 subchannels), the module numbering need not be contiguous; 4, 0 and 0224, 0220 and 0314 is a typical possibility for five DSC-II subchannels. Transmissions to and from the DSC and memory may be under internal interlace control or, when so equipped, under external interlace control.

Internal Interlace

During an internal interlace transmission, the DMC controls the interlacing operation in the following order:

1. Access interlace word. The DMC accesses the interlace word assigned to the requesting subchannel.
2. Process interlace word. The DMC increments the 15-bit address portion of the word and decrements the 19-bit word count.
3. Test for zero and set indicator. Next, the DMC tests the word count for zero and if it is zero, sets an indicator in the pertinent DSC.
4. Restore. The DMC then places the new word count/address values back into memory using the assigned address of requesting subchannel.
5. Access/store as requested. The DMC accesses or stores the transmitted word as requested using the incremental address (see above).
6. Stop or continue. The DSC checks its zero count indicator and
   a. if zero and working on the even interlace word, the DSC continues operation using the odd interlace word.
   b. if zero, working on the odd interlace word and the cycle bit is set, the DSC continues using the even interlace word.
   c. if zero, working on the odd interlace word and the cycle bit is reset, the DSC terminates the operation on a DSC-II or responds as required by the function control on a DSC-I.
   d. if not zero, the DSC returns operation to the DMC to continue at 1 (above).

Note that the first address used is the "address specified plus one" and the first word count is the "word count specified minus one". In particular, an initial word count of zero causes a 256-word block to be transmitted.

External Interlace

During transmissions utilizing external interlace control, the interlacing operation proceeds as described above except that when the DSC is activated, the DSC with external interlace (EIN) requests the DMC to access the desired interlace control word. The interlace control word is sent to the EIN. Thereafter, data transmissions to and from the DSC and memory utilize the interlace address and word count supplied by the EIN.

Data transmissions using the EIN require only one cycle while those data transmissions using internal interlace require three cycles. Should a transmission result in the EIN detecting a zero-word-count condition, the DSC-EIN will restore the external interlace word and will proceed according to 6 (above). Any termination of a DSC operation prior to zero word count due to any externally derived halt signal also causes a restoring of the EIN interlace control word.

DSC PROGRAMMING

An EOM-POT sequence selects, alerts, and controls the subchannel; an EOM-SKS sequence selects and tests the condition of the subchannel. The "select" EOM has the form:

0	02	7	0	C	0	DSC number
0 2	3 8	9 11	12	13 14	15	16 23

Bit positions 16-23 contain the DSC number being alerted; these numbers are the even numbers from 0 through 376B. The C field (bits 13, 14) specifies one of three modes to which the DSC is alerted, as indicated on the following page.

C	Effect
00	The subchannel decodes the lower 12 bits (12–23) of the "POTted" word to follow as the lower 12 bits of a buffer control mode EOM. For DSC-I, this will select a device with the unit address field, set the character/word count, specify binary or BCD format, forward or reverse, and leader or no leader. For DSC-II, the 12 bits activate the subchannel and select the proper unit (if more than one is attached to the DSC).
01	The subchannel decodes the lower 12 bits of the "POTted" word as the lower 12 bits of an input/output control mode EOM. If bits 18 through 23 are zero, the "POTted" word to follow addresses the selected DSC. For DSC-I, these bits perform such functions as rewind tape, space paper, etc. For DSC-II, these bits perform such functions as required by the selected device attached to the DSC.
10	The subchannel decodes the lower 12 bits of the "POTted" word to follow for controlling the interlace and interrupts. The control type EOM should precede the buffer control EOM. For DSC-I the form is:

00	FC	ER	ZC	CY	EO
12–17	18–19	20	21	22	23

FC is a 2-bit function code similar to the TMCC/DACC terminal function codes. The remaining bits function as described below for DSC-II. For DSC-II, the form is:

00	0	0	ER	ZC	CY	EO
12–17	18	19	20	21	22	23

Bit Position	Function
20	A 1 in the ER bit arms the end-of-record interrupt for this channel.
21	A 1 in the ZC bit arms the zero-word-count interrupt.
22	A 1 in the CY bit (cycle) sets the cycle mode such that the interlace will switch from the odd-numbered interlace word back to the even-numbered interlace word when the word count in the odd-numbered word is reduced to zero. If ZC and CY are both set to 1, a zero-word-count interrupt is generated each time the interlace switches (to either word – even or odd). If CY is set to 0, the interlace will not proceed after the word count in the odd-numbered word is zero; and a zero-word-count interrupt occurs only when the word count in the odd-numbered interlace word is zero.
23	A 0 in the E/O bit selects the even-numbered interlace word as the first interlace word in a transmission; note that when starting on the even-numbered word, the interlace always switches to the odd-numbered word for further control when the word count in the even-numbered word goes to zero. A 1 in E/O sets the odd-numbered interlace word as the first interlace word in a transmission; the interlace operation ceases when the word count in the odd-numbered word reaches zero unless the CY bit is set to 1.

Terminating DSC Input Output

Once the cycle bit has been set, the interlace continues to cycle back and forth between the even/odd interlace words. An EOM-POT sequence is used to terminate the cycle. The select EOM is:

0	02	71	0	DSC number
0–2	3–8	9–14	15	16–23

The lower 12 bits of the following "POTted" word must be:

0200
12–23

The interlace terminates the next time the count reaches zero in the odd interlace word.

For example, to terminate the cycle on DSC 4, use the sequence: EOM 71004B, POT A (where location A contains the value 200B).

The SKS to test subchannels has the form:

0	40	7	Test	0	Unit
0–2	3–8	9–11	12–16	17	18–23

A select EOM with bits 13 and 14 coded 00 permits the following SKS to be directed to the subchannel or to the device attached to it. The Unit field specifies the device to be tested; the Test field is defined for the particular device. When testing the subchannel, the Unit field is set to 00B. The Test field contains the same testing format as SKS for testing a TMCC.

For example, to test DSC 4 for error, use the sequence:

```
EOM 70004B
SKS 71000B
```

MEMORY INTERFACE CONNECTION

Once a computer is equipped with a multiple access to memory feature, one or more memory interface connections (MIC) can be attached. The MIC is a general interface between the computer and the outside world that allows special devices to be connected to the computer. The MIC converts between the 4-volt logic levels used in the computer and the 8 volts used outside. It preserves the integrity of the memory by generating the parity of incoming data words. It will also check the parity of words read from memory to indicate memory failures. If incoming data is supplied with parity, the MIC will check for odd parity as it generates the internal memory parity and respond with a signal that indicates if the transmission was correct. The device that is connected to the MIC must store both the data and the address until the transmission to/from memory is completed.

5. OPERATOR CONTROLS

CONTROL PANEL

The XDS 940 Computer provides a control panel (see Figure 8) for operator control. The control panel connects directly to the central processor, contains switches for operations, and displays the contents of operational registers. The registers displayed on the control panel directly reflect the contents of the hardware registers. If the operator changes or clears a display, the contents of the actual register also change identically.

POWER

The POWER switch turns the computer power system on or off, and is lighted in the on condition.

I/O DISPLAY SELECT

This eight-position, thumbwheel switch selects the channel from which the UNIT address and ERROR indicators are displayed in the INPUT-OUTPUT lights. It also selects the channel to be used in a "fill" operation.

FILL

The operator has the option of four input media to initially load or "fill" the computer. The pair of three-position, spring-loaded, center-return, toggle FILL switches are labeled: PAPER TAPE, MAG TAPE, CARDS, and DRUM. For example, to select and initiate filling from paper tape, set the first toggle switch to PAPER TAPE and release.

OVERFLOW

This display shows the status of the overflow indicator.

HALT

This indicator displays the current status of the halt flip-flop. If the computer executes a HALT (HLT) instruction, the halt flip-flop is set and the HALT indicator is turned on. Placing the RUN-IDLE-STEP switch in IDLE clears the halt flip-flop and turns off the HALT indicator.

INTERRUPT ENABLED

This indicator is on if the interrupt system is enabled, and is off if the system is disabled. The switch below this indicator allows the operator to enable the interrupt system. In the ENABLE position, the switch enables the interrupt system regardless of program operations; in the COMPUTER position, the switch allows the program to enable or disable the interrupt system. The switch is stationary in the COMPUTER position and momentary in the ENABLE position.

MEMORY PARITY

If an operand, instruction, or access from memory encounters a parity error, MEMORY PARITY lights. When the switch below the indicator is in the HALT position, the computer enters the idle state whenever a memory parity error occurs. Setting the switch to CONTINUE clears the MEMORY PARITY indicator and the computer continues

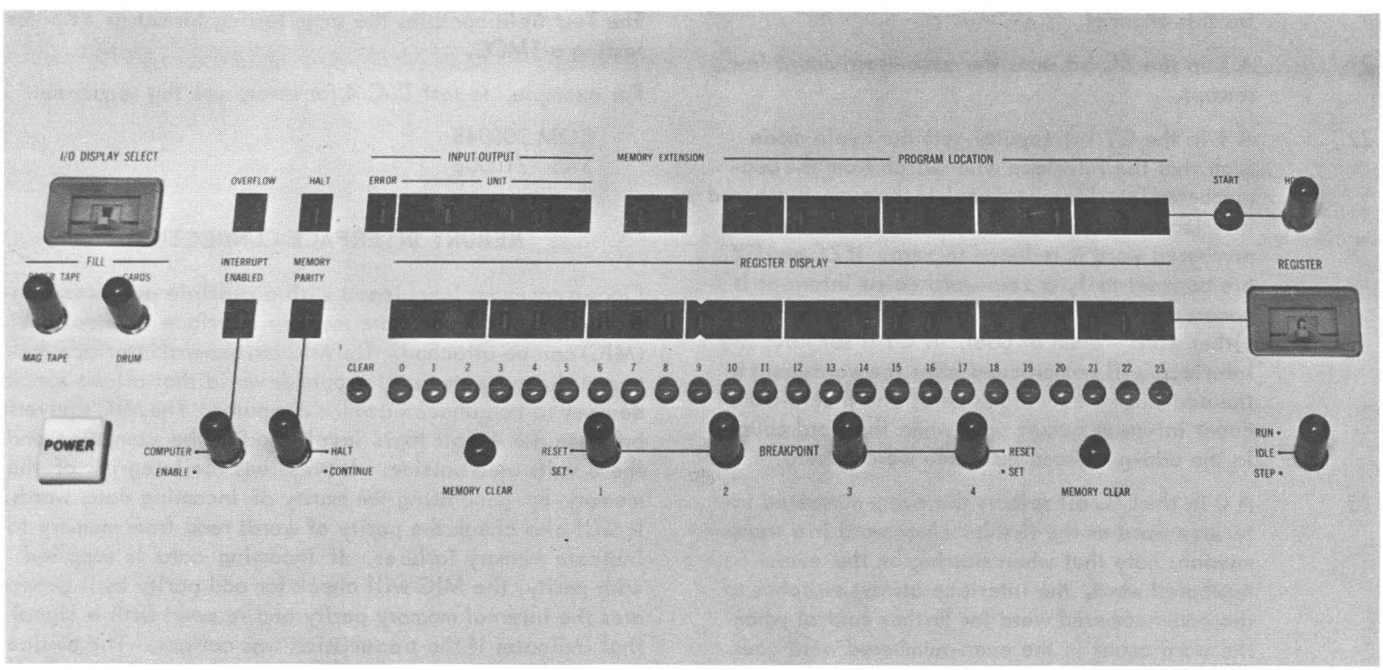


Figure 8. XDS 940 Control Panel

normal operation. If the switch is in the CONTINUE position when a memory parity error occurs, the computer ignores the error.

If the memory parity interrupt option is present and the MEMORY PARITY switch is in the HALT position when a memory parity error occurs, the computer does not halt; instead, it activates the appropriate memory parity interrupt level (see Section 2, "Memory Parity Interrupts").

INPUT-OUTPUT

The UNIT lights contain the input/output unit address of the peripheral device currently connected to the selected channel. The ERROR light reflects the current status of the channel error indicator. The current setting of the I/O DISPLAY SELECT thumbwheel switch selects the channel to be displayed.

MEMORY EXTENSION

There are two memory extension indicators. The left indicator is lighted when extend memory register EM3 does not contain the value 3; the right indicator is lighted when extend memory register EM2 does not contain the value 2.

PROGRAM LOCATION

This display consists of 14 binary indicators that show the current contents of the P register (program counter). When the RUN-IDLE-STEP switch is in IDLE, the indicators in this display contain the virtual address of the next instruction to be executed. This display (and thus the P register) may be changed by entering a BRU instruction into the C register with the set buttons and then executing the BRU instruction (see "Operating Procedures").

START

This switch is used to initialize the normal mode of the computer. It also resets all channel registers and indicators to zero, clears and disarms all interrupt levels, disables the interrupt system, sets the extend memory register EM3 to 3, sets the extend memory register EM2 to 2, and clears the P register, overflow indicator, MEMORY PARITY error indicator, and C register to zeros. The RUN-IDLE-STEP switch must be in IDLE and the REGISTER SELECT switch must be at C when pressing this switch.

HOLD

Placing the HOLD switch in the up position causes the current contents of the program counter (P register) to be held and prevents it from counting. At this time, the operator can insert instructions into the C register and execute them without stepping the program counter. When the HOLD switch is in the down position, the program counter is automatically incremented by 1 as each instruction is executed.

REGISTER DISPLAY

This display consists of 24 binary indicators that show the contents of the register selected by the REGISTER switch. With the RUN-IDLE-STEP switch in IDLE, pressing the CLEAR pushbutton clears the selected register to all zeros. By pressing the pushbuttons beneath selected indicators, the operator may enter any desired configuration of bits into the selected register. If the operator clears or changes this display, the actual contents of the selected register change identically.

REGISTER

This four-position, rotary switch selects the internal register to be displayed in REGISTER DISPLAY. The selectable registers are:

C register (arithmetic and control)

A register (main accumulator)

B register (extended accumulator)

X register (index)

MEMORY CLEAR

These two switches are used to clear core memory to zeros. To clear the first 16K words of memory, press the START switch and then press both MEMORY CLEAR switches simultaneously. To clear from 16K through 24K-1, set the extend memory registers EM2 to 4 and EM3 to 5, then press these two MEMORY CLEAR switches simultaneously. To clear from 24K to 32K-1, set the extend memory registers EM2 to 6 and EM3 to 7, then press both MEMORY CLEAR switches simultaneously. A monitor-mode program is required to clear locations 32K through 64K-1.

BREAKPOINT

The normal-mode or monitor-mode program may detect the status of these four switches by using a breakpoint test (see Section 3, "Breakpoint Instructions"). The switches, labeled RESET and SET, control predetermined options within the program.

RUN-IDLE-STEP

This three-position, toggle switch has two stationary positions (RUN and IDLE) and a spring-loaded, momentary position (STEP). With this switch in the RUN position, instruction execution occurs automatically at computer speed.

When this switch is placed in the IDLE position, the computer "idles" immediately after obtaining an instruction from memory. If at the same time, the REGISTER switch is in position C, the next instruction to be executed is shown in REGISTER DISPLAY. Also, if the Halt flip-flop has been set by a HALT (HLT) instruction, moving the RUN-IDLE-STEP switch from the RUN position to the IDLE position resets the Halt flip-flop.

Moving the switch to the STEP position causes the computer to execute the current contents of the C register, load the

C register with the next instruction in sequence, and automatically return to an idle state. The RUN-IDLE-STEP switch must be allowed to return to the IDLE position before it can be activated again to execute the next instruction.

OPERATING PROCEDURES

The following are recommended control console operations to accomplish common computer functions.

TURN COMPUTER ON

1. Set the RUN-IDLE-STEP switch to IDLE.
2. Press POWER switch.

LOAD PROGRAM WITH LOADING SYSTEM

Refer to the operating procedures furnished with the particular assembler, compiler, monitor, diagnostic, or utility system being used.

LOAD PROGRAM WITH FILL SWITCH

1. Set up the selected input device with the input program. The initial portion of the program contains the "bootstrap" (the short-load program).
2. Set the RUN-IDLE-STEP switch to the IDLE position.
3. Press the START switch.
4. Set the RUN-IDLE-STEP switch to the RUN position.
5. Set the I/O DISPLAY SELECT switch to W.
6. Press one of the four FILL switches. This will cause a WIM 2 instruction (03200002B) to be inserted into the instruction register and will load the index register with 77777771B (-7). Depending on which switch is pressed, activation of one of the following four devices on channel W will occur:

   Paper tape reader 1 (unit address 04B)

   Card reader 1 (unit address 06B)

   RAD file 1 (unit address 26B)

   Magnetic tape unit 0 (unit address 10B)

   The FILL switch also prepares the channel to operate in the forward, binary, four characters per word mode.

A boostrap program must be in position to be read as the first input from the device. A typical bootstrap program is:

Location	Instruction
2B	WIM 12B, 2
3B	BRX 2
4B	LDX 11B
5B	WIM 0, 2
6B	SKS 21000B
7B	BRX 5
10B	(First instruction)
11B	(Starting address with indirect address tag)

The WIM 2 instruction that is forced into the C register stores the first word of the bootstrap program in location 2. The computer then executes the contents of location 2. The index register, which contains -7, modifies the WIM in 2. The effective address of the WIM is then 3 so that the second word is stored in 3. This word is a BRX back to the WIM.

These two instructions then load the remainder of the bootstrap program. The remaining six words can be those needed for the specific loading that is to be done. The one shown loads a record of up to 16K words. The channel active test in location 6 skips when the end-of-record has been reached. In "bootstrapping" from paper tape to magnetic tape, the record may be of any length. From cards, the record is 40 words.

EXECUTE PROGRAM

1. Set the RUN-IDLE-STEP switch to IDLE.
2. Set the REGISTER switch to C.
3. Press CLEAR and enter a BRU to the program starting location into REGISTER DISPLAY, using the set buttons. Format of the instruction is:

   000 000 001 0xx xxx xxx xxx xxx

   BRU (000 000 001) Program starting location (0xx xxx xxx xxx xxx)
4. Set the RUN-IDLE-STEP switch to RUN. The computer then executes the BRU and continues instruction execution at computer speed. Or, set the RUN-IDLE-STEP switch to STEP and release the switch. The computer executes the BRU and returns to the idle state with the contents of the first instruction of the program displayed in REGISTER DISPLAY, and the virtual address of the first instruction of the program displayed in PROGRAM LOCATION. The operator may continue to cause the computer to execute instructions in this manner by repeatedly setting the RUN-IDLE-STEP switch to STEP, allowing the switch to return to IDLE each time. This process is called "stepping" instructions.

INSPECT MEMORY CONTENTS

1. Set the RUN-IDLE-STEP switch to IDLE.
2. Set the REGISTER switch to C.
3. Press CLEAR and enter a BRU to the virtual address of the memory location to be examined into REGISTER DISPLAY, using the set buttons. Format of the instruction is:

   000 000 001 0xx xxx xxx xxx xxx

   BRU (000 000 001) Memory location (0xx xxx xxx xxx xxx)
4. Set the RUN-IDLE-STEP switch to STEP and release the switch. PROGRAM LOCATION now contains the 14-bit virtual address of the location to be inspected and REGISTER DISPLAY contains the 24-bit contents of the location.
5. To inspect other memory locations, repeat steps 3 and 4 above.

MODIFY MEMORY CONTENTS

1. Set the RUN-IDLE-STEP switch to IDLE.
2. Set the REGISTER switch to A.
3. Press CLEAR and enter the desired configuration into the A register, using the set buttons below REGISTER DISPLAY.
4. Set the REGISTER switch to C.
5. Enter 035 XXXXX into REGISTER DISPLAY, using the set buttons. (035 is the octal instruction code for STORE A, and XXXXX is the virtual address of the memory location to be changed.)
6. Set the RUN-IDLE-STEP switch to STEP and release the switch. The computer executes the STORE A instruction and returns to the idle state.

INSPECT/MODIFY REGISTER CONTENTS

1. Set the RUN-IDLE-STEP switch to IDLE.
2. Set the REGISTER switch to the desired register (A, B, C, or X). The contents of the selected register are immediately displayed in REGISTER DISPLAY and may be changed by pressing CLEAR and inserting a new configuration with set buttons.
3. Set the REGISTER switch back to C before placing the RUN-IDLE-STEP switch into RUN or STEP.

CLEAR HALT CONDITION

1. Set the RUN-IDLE-STEP switch to IDLE.
2. To continue with the displayed instruction, set the RUN-IDLE-STEP switch to RUN (for automatic operation) or to STEP for single-stepping.

6. PERIPHERAL EQUIPMENT

This section describes some of the input/output devices that can be attached to a channel, specifies the EOM and SKS instructions for each device, and provides standard programming approaches for hardware conditions peculiar to each device. In the programming examples, all octal integers are followed by the letter "B", unless otherwise specified; decimal integers are not followed by the letter "B".

TYPEWRITER INPUT/OUTPUT

The electric input/output typewriter is used for operator control, error or status messages, and similar functions. The typewriter has no ready test; thus, it is considered to be always ready.

TYPEWRITER INSTRUCTIONS

The typewriter instructions to follow are coded without interlace, using channel W at 4 characters/word, on unit 1.

READ KEYBOARD EOM 2601B

This instruction connects the typewriter to the channel, turns on the typewriter (lights the input light), and initializes the channel to assemble 4 characters/word.

When a typewriter input operation immediately follows typewriter output, the program must allow 40 milliseconds (22,840 computer cycles) after the channel disconnects before executing a READ KEYBOARD instruction. Otherwise, the last character transmitted to the typewriter may reappear as the first character read back into the channel.

TYPE TYPEWRITER EOM 2641B

This instruction connects the typewriter to the channel, turns on the typewriter, and initializes the channel to output 4 characters/word.

TERMINATING TYPEWRITER INPUT/OUTPUT

Since the typewriter is not a record-oriented device, it provides no terminating signals. Thus, if single-word transmission is used for typewriter input, the program must disconnect the typewriter at the end of an input with a DISCONNECT CHANNEL W (DISW) instruction. If single-word transmission is used for typewriter output, the program must terminate the output operation with a TERMINATE OUTPUT ON CHANNEL W (TOPW) instruction. If the channel unit address register is not cleared after a typewriter input or output, the CHANNEL W ACTIVE TEST (CATW) will not cause the computer to skip an instruction. If typewriter input/output is accomplished using interlace, the interlace control automatically terminates input/output (clears the unit address register in the channel).

ERROR CONDITIONS

The typewriter does not generate error signals, but if an input/output parity error or data overrun (character rate error) is detected by the channel, the error flip-flop in the channel is set and the INPUT/OUTPUT ERROR indicator on the control panel is turned on.

Typewriter output example:

This program types out an 8-word message. The program is written as a closed subroutine that uses the zero-word-count interrupt level, channel W with interlace, and typewriter 1.

Location	Instruction	Address	Comments
RKB	ZRO		This instruction is an assembler directive, used here as a convenient way to reserve the entry location for subroutine use.
	CLA STA	 SWICH	This pair of instructions clears the location called SWICH. SWICH is later used to indicate to the main program that output is complete.
	EOM	42641B	This instruction alerts the channel W interlace, connects typewriter 1 to channel W for output, and specifies the 4 characters/word mode.
	EOM	15200B	This instruction specifies extended mode, arms the zero-word-count interrupt level, and specifies output terminal function IOSD.
	POT	WRITE	This instruction sends the word count and starting address in location WRITE to the channel.
	BRR	RKB	This instruction branches back to the main program.

(continued)

Location	Instruction	Address	Comments
WRITE	DATA	403720B	The word in WRITE specifies that eight words will be output from memory beginning in location 3720B. According to output function 01 (IOSD) when the word count equals zero during the transmission, the typewriter is disconnected when the last character is out, at this time, the zero-word-count interrupt signal occurs.

The main program is processed while the output operation is being performed by the channel. When finished with the output, an interrupt signal will be transmitted to interrupt level 31B.

31B	BRM	OKAY	This instruction, placed in location 31B, branches and marks to location OKAY elsewhere in memory.
⋮			
OKAY	ZRO		This instruction saves the entry location.
	MIN	SWICH	This instruction increments location SWICH, which is used as an indicator for the main program.
	BRI	OKAY	This instruction branches to the main program and clears the active interrupt, level 31B.

Typewriter input example:

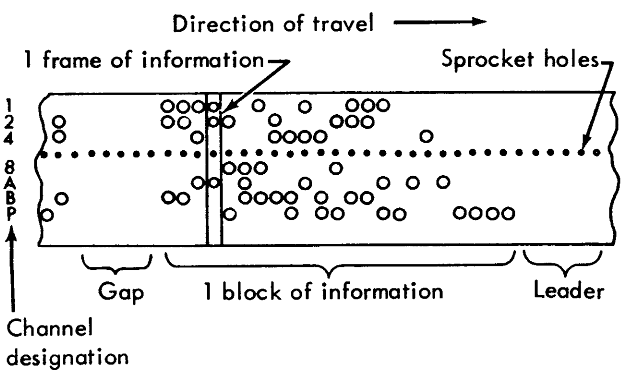
The operator is to input four control characters; the subroutine is assumed to have been entered under program control. The terminal interrupts are not requested in this example.

Location	Instruction	Address	Comments
TYP	ZRO		This assembler directive reserves the subroutine entry location.
	EOM	42601B	This instruction alerts the channel W interlace, connects typewriter 1 to channel W for input, and specifies the 4-characters/word mode. Also, the input request light on the keyboard is turned on.
	EOM	14200B	This instruction specifies the extended mode, no interrupts, and specifies input terminal function IOSD.
	POT	CHARS	This instruction transmits the word count and starting address to the channel.
	CATW		This instruction tests for channel W active. If the channel is active when CATW is executed, the next instruction in sequence is executed. If the channel is inactive, the next instruction is skipped and the following one is executed. The octal configuration of this instruction is: 0 40 14000.
	BRU	*-1	This instruction branches back to the CATW instruction. The asterisk and accompanying signed integer in the address field is an assembler declaration for the indicated number of locations prior to or following the current one. Plus indicates following.
	BRU	CHECK	This instruction branches to an assumed routine to determine what characters were typed in.
CHARS	DATA	47640B	The word in CHARS specifies that one word can be input into location 7640B. Only one word will be accepted before the channel disconnects and goes inactive. The zero word count causes the channel to disconnect from the typewriter and turns off the input request light.

PAPER TAPE INPUT/OUTPUT

PAPER TAPE FORMAT

The paper tape uses six hole positions for information and one for odd parity check in each frame. The paper tape is one inch wide, with ten frames of information per inch in the direction of travel. Information is organized on the tape in blocks. A block is any number of information frames, set off by a gap (in which only sprocket holes are punched) at either end. Gap in front of the first block is called "leader."



PAPER TAPE READER

The paper tape reader is primarily used for loading programs and data into memory. The reader is always ready for operation and no ready test is required. Before executing the EOM instruction to read a tape, the tape must be loaded into the reader. The loading procedure is:

1. Place the tape actuator into the LOAD position.
2. Insert the tape (from left to right) into the tape guide, with channel P toward the operator. (If a spool of tape is used, mount the spool on the spooler and thread the tape into the take-up spool.)
3. Place the tape actuator in the RUN position.

READ PAPER TAPE EOM 2604B

This instruction connects paper tape reader 1 to channel W, starts the tape moving, and transmits a block of information (1 character at a time) to the buffer. The reader ignores leader and, unless otherwise instructed by another EOM, stops within one frame of gap, generates an end-of-record signal, and disconnects from the channel buffer (clears the unit address register).

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 with a DISW instruction, and then start again to read the remainder or another portion of the block. However, the paper tape reader must not be restarted until at least 30 milliseconds (approximately 17,130 computer cycles) have elapsed following the previous read operation. Since the paper tape reader stops between frames, no frame is missed between subsequent read operations.

Terminating Paper Tape Input

Once a paper tape read operation is started, the paper tape reader should not be disconnected (by DISW) until at least 4 characters have been read, to prevent damaging the read mechanism. Also, if only a portion of a block is to be read in the compatible mode, DISW must be executed within 0.3 millisecond (approximately 171 computer cycles) after the last character is read. Otherwise, characters continue to enter the channel and a data overrun (character rate error) occurs. (The program may also store the unwanted remainder of the record into an unused portion of memory. When the reader disconnects, after reading the last character, an end-of-record interrupt occurs if the interrupt level is armed and enabled.)

Error Conditions

If a parity error or data overrun occurs during a paper tape read operation, the channel error flip-flop is set and the INPUT/OUTPUT ERROR indicator on the computer control panel is turned on.

PAPER TAPE PUNCH

The paper tape punch is primarily used for punching programs and/or data to be later loaded back into memory. The punch is always ready for operation and no ready test is required. Before executing the EOM to punch a tape, the operator should determine if there is enough tape on the supply reel for the punching operation and that the tape is properly threaded. For extensive punching operations, the tape should be threaded onto a take-up reel. After each roll of tape has been punched, the operator must empty the chad box and brush all loose chad from the tape guide. Otherwise, the punch may jam during a punching operation.

If the toggle switch on the punch panel is placed in the RUN position, the punch motor runs continuously. If the switch is in the AUTO position, the punch motor is turned on only when the punch is addressed by the program (with an automatic delay to allow the motor to reach punching speed) or when the FEED button on the punch panel is pressed. Tape leader may be manually punched by pressing the FEED button until the desired amount of leader is produced. The following punch tape instructions are coded for channel W, using unit 1 at 4 characters/word, without interlace.

PUNCH PAPER TAPE WITHOUT LEADER EOM 2644B

This instruction connects the paper tape punch to the channel, starts the punch motor (if not already on), and initializes the buffer to output 4 characters/word. Since bit position 13 contains a 1, no leader is generated before punching the first frame.

PUNCH PAPER TAPE WITH LEADER EOM 644B

This instruction is identical to PUNCH PAPER TAPE WITHOUT LEADER except that bit position 13 contains a 0, to

Paper tape input example:

This program will read a block of 64 characters from paper tape. The 4-characters/word format makes the input 16 words. The routine is written as a closed subroutine that uses the zero-word-count interrupt level, channel W with interlace, and paper tape reader unit 1.

Location	Instruction	Address	Comments
RPT	ZRO		This assembler directive saves a place for the entry location.
	CLA STA	 SWICH	This pair of instructions clears location SWICH, which will be used as an input-finished indicator.
	EOM	42604B	This instruction alerts the channel W interlace, connects paper tape reader 1 to channel W, and specifies the 4-characters/word format.
	EOM	15200B	This instruction specifies the extended mode, arms the zero-word-count interrupt level, and specifies input terminal function IOSD.
	POT	REED	This instruction transmits the word count and starting address to the channel.
	BRR	RPT	This instruction branches back to the main program for processing while the input operation is in progress.
REED	DATA	1003720B	The word in REED specifies that input into memory begins in location 3720B and that 16 words will be read before the operation is completed.

When the word count equals zero, the zero-word-count interrupt occurs, the channel disconnects, and the paper tape reader is stopped before reading the next frame. The zero-word-count interrupt occurs at level 31B.

Location	Instruction	Address	Comments
31B	BRM	FNISH	This instruction, in location 31B for this example, branches and marks to location FNISH.
⋮			
FNISH	ZRO		This assembler directive saves the entry location.
	MIN	SWICH	This instruction sets an input-finished switch for use by the main program.
	BRI	FNISH	This instruction branches back to the main program and clears interrupt level 31B from the active state.

A test to the channel, CETW, for parity error during the read operation can be made before the BRI instruction.

specify that the punch generate approximately 1 inch of leader preceding the first frame. The PUNCH PAPER TAPE WITH LEADER instruction may be used to form separate blocks of information on a single tape, when successive punching operations are executed.

Terminating Paper Tape Output

The paper tape punch continues to punch as long as it receives characters from the channel, regardless of the infrequency of transmission. The punch operates at 60 characters per second, asynchronously. If the channel does not supply characters at the punch fast enough for operation at 60 cps, the punch waits for each character, losing no data and creating no blank frames, unless so instructed by a PUNCH TAPE WITH LEADER instruction. Thus, the program must disconnect the tape punch at the end of the output operation. Otherwise, the channel unit address register is not cleared, and the computer will not skip the next instruction when CATW is subsequently executed. If the punch operation is accomplished under interlace control, a TERMINATE OUTPUT ON CHANNEL W (TOPW) instruction is automatically generated. If single-word transmission is used, the program must contain the TOPW instruction.

The paper tape punch does not automatically produce gap after punching a block of information. If gap is desired, the operator may depress the FEED button to produce the desired gap. Also, the program may instruct the punch to produce a 1-inch gap by executing a PUNCH TAPE WITH LEADER instruction, followed immediately by a TOPW instruction.

Error Conditions

If a parity error occurs during a paper tape punch operation, the channel error flip-flop is set and the INPUT/OUTPUT ERROR indicator on the computer control panel is turned on.

Paper tape output example:

This program will punch one block of 20 words beginning in location 2000B. A 1-inch leader precedes the block. The routine is a closed subroutine that uses the zero-word-count interrupt level, channel W with interlace, and paper tape punch unit 1.

Location	Instruction	Address	Comments
PPT	ZRO		This assembler directive saves a place for the entry location.
	CLA		This pair of instructions clears a switch location used as an indicator to the main program for completion of the punch operation.
	STA	WHERE	
	EOM	40644B	This instruction alerts the channel W interlace, connects channel W to paper tape punch 1 and specifies the 4-characters/word mode. The instruction also specifies that leader is to be punched and that the punch motor is to be turned on (if not already on).
	EOM	15200B	This instruction specifies the extended mode, arms the zero-word-count interrupt level, and specifies output terminal function IOSD.
	POT	PUN20	This instruction transmits the word count and starting address of the transmission to the channel.
	BRR	PPT	This instruction branches back to the main program.
PUN20	DATA	1202000B	The word in PUN20 specifies that 20 words will be output from memory to the punch beginning at location 2000B.

According to output function 01, when the word count equals zero during the transmission, the zero-word-count interrupt occurs. The last word has not been fully transmitted at this time; when it is and the output is complete, the channel disconnects. When the zero-word-count interrupt occurs, the following instructions are executed.

31B	BRM	END	This instruction branches and marks to location END.
⋮			
END	ZRO		This assembler directive reserves a location for subroutine entry.
	MIN	WHERE	This instruction increments location WHERE to indicate that the output operation is complete.
	BRI	END	This instruction clears interrupt level 31B and branches back to the main program.

CARD INPUT/OUTPUT

CARD FORMAT

Two formats are available for reading and punching 80-column cards: Hollerith and binary. Hollerith format, as shown in Figure 9, consists of up to 80 Hollerith-coded characters per card, with each character represented by a single column. Thus, a card may represent up to 80 characters (20 words at 4 characters/word) in Hollerith format.

Binary format consists of two 6-bit characters per column. The top 6 rows (12-3) of column 1 form the first character (with the most significant bit in row 12), the bottom 6 rows (4-9) form the next character (with the most significant bit in row 4). Thus, a single card may represent up to 160 characters (40 words at 4 characters/word) in binary format.

CARD READER

Before initiation of a card read operation, the card reader should be loaded and tested as follows:

1. Loading procedure:
   a. Press card reader POWER ON switch.
   b. Place cards into hopper (face down with row 12 towards the operator) and place plastic weight on the cards.
   c. Press card reader START switch.
2. Testing procedure:
   a. Test channel (channel active test).
   b. Test card reader (card reader ready test).

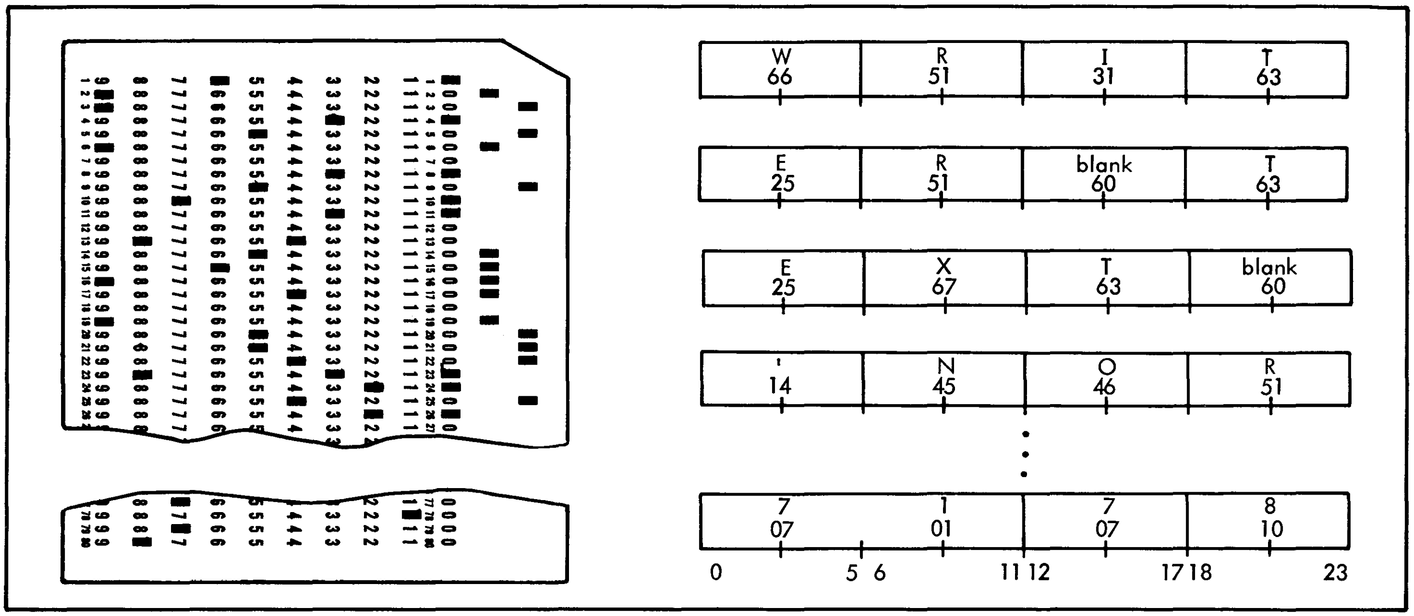


Figure 9. Card Read Into Memory in Hollerith

Card Reader Tests

The card reader tests to follow are coded for channel W, using unit 1.

CARD READER READY TEST SKS 12006B
(Skip if card reader ready)

The card reader is ready to feed and read when all of the following conditions exist:

1. Card reader POWER ON switch is on
2. Hopper is not empty
3. Stacker is not full
4. Feed mechanism is operating properly
5. Read mechanism is operating properly
6. Card reader START switch has been pressed
7. No feed or read cycle is in process

If the card reader is ready when the card reader ready test is executed, the computer skips the next instruction in sequence and executes the following instruction. If the card reader is not ready, the computer executes the next instruction in sequence (does not skip). This ready test should be made before each EOM instruction that initiates a read cycle.

FIRST COLUMN TEST SKS 14006B
(Skip if not first column)

This test determines if the first column is about to be read by the card reader. Since the time elapsing between the execution of a card reader EOM and the reading of the first column is approximately 85 milliseconds (48,450 computer cycles), the computer can perform other operations during this time. If the first column test instruction is executed less than 1.2 milliseconds (approximately 685 computer cycles) before the first column is due to be read, the computer skips the next instruction in sequence and executes the following instruction. If the first column test is executed 1.2 milliseconds (or more) before the first column is due to be read, the computer executes the next instruction in sequence (does not skip).

CARD READER END-OF-FILE TEST SKS 11006B
(Skip if not end-of-file)

This test determines if an end-of-file (EOF) condition exists for the card reader. The EOF condition exists when the hopper is empty and the EOF ON indicator switch is lighted. (The END OF FILE indicator is also lighted when the EOF condition exists.) If the EOF condition exists, the computer executes the next instruction in sequence (does not skip), and the EOF condition continues until the operator adds cards to the hopper or resets the EOF ON switch. If the EOF condition does not exist, the computer skips to the next instruction in sequence and executes the following instruction.

Card Reader Instructions

If the card reader is in a ready condition when the read card EOM is executed, the reader reads 1 card (column by column, starting with column 1), transmits 80 Hollerith (or 160 binary) characters to the channel, generates an end-of-record signal, and waits for the next EOM. The card reader instructions to follow are coded without interlace, using channel W at 4 characters/word, for unit 1.

READ CARD IN DECIMAL (Hollerith) EOM 2606B

This instruction alerts the card reader, causes a card to feed from the hopper, and specifies the Hollerith format. As each column is read, it is translated to XDS internal code.

Card input example:

This program reads one card in Hollerith mode. It is written as a closed subroutine that uses channel W with interlace, card reader unit 1, the zero-word-count interrupt level, and the end-of-record interrupt level.

Location	Instruction	Address	Comments
RCD	ZRO		This assembler directive saves a location for the subroutine entry.
	SKS	12006B	This instruction is the card reader ready test for card reader 1 on channel W. If not ready, the instruction is executed. If the card reader is ready, the next instruction is skipped and the following one is executed.
	BRU	*-1	This instruction branches back to the card reader ready test. An exit to a not-ready corrective routine can be put here.
	EOM	42606B	This instruction alerts the channel W interlace, connects card reader 1 to channel W, and starts a card moving toward the read station. The 4-characters/word, Hollerith mode is specified.
	EOM	17200B	This instruction specifies the extended mode, arms the end-of-record and zero-word-count interrupt levels, and specifies input terminal function IOSD.
	POT	READ	This instruction transmits the word count and starting address to the channel.
	BRR	RCD	This instruction branches back to the main program.
READ	DATA	1203720B	The word in READ specifies that a record will be read into memory beginning at location 3720B. A 20-word limit is specified.

The main program is processed while the card read operation is being performed by the channel. When finished with the input, an interrupt signal will be transmitted to interrupt level 31B, the zero-word-count interrupt level for channel W. Should the card reader become disconnected because of a READ CHECK or FEED CHECK before 20 words are read, an interrupt signal will be transmitted to interrupt level 33B, the end-of-record interrupt level.

31B	BRM	TEST	These instructions, placed in locations 31B and 33B for this example, branch and
33B	BRM	TEST	mark to location TEST.
:			
TEST	ZRO		This instruction saves a location for the routine entry.
	CETW		This instruction tests for an error on channel W; its octal configuration is: 0 40 11000.
	BRM	ERR	This instruction is executed if there is an error on channel W. It is assumed that ERR is the entry to a corrective subroutine.
	BRI	TEST	This instruction is executed if no error is detected; it returns control to the main program and clears the active interrupt level.

READ CARD IN BINARY EOM 3606B

This instruction alerts the card reader, causes a card to feed from the hopper, and specifies the binary format. As each column is read, it is transmitted as two 6-bit binary-coded characters.

The reading mode may be changed between card columns by executing EOM instructions with the appropriate format code. This provides a means of reading cards that have some fields punched in Hollerith and others in binary. At times, only the first portion of a card has information required by the program. In order to save the computer time required to process the unwanted information, the reader may be instructed to skip the remainder of the card.

SKIP REMAINDER OF CARD BEING READ EOM 12006B

This instruction causes the reader to stop transmission of characters to the channel. The remaining characters are not checked for validity, but a read check, feed check, or end-of-record condition still causes an end-of-record interrupt and disconnect the card reader from the channel.

Error and Disconnect Conditions

If the card reader has been instructed to read a card, the card reader responses to error/disconnect conditions are as follows:

Condition	Card Reader Response
1. Feed malfunction	a. Disengage card reader motor
	b. Turn on FEED CHECK indicator

Condition	Card Reader Response
	c. Turn on NOT READY indicator
	d. Set error flip-flop in channel (test with CETW)
	e. Disconnect card reader from channel (clear unit address register); channel then generates end-of-record interrupt signal
2. Read malfunction	Turn on READ CHECK indicator (other responses are identical to feed malfunction)
3. Validity error	a. Turn on VALIDITY CHECK indicator
	b. Set error flip-flop in channel
4. End of card (end of read cycle)	a. Disengage card reader motor
	b. Disconnect card reader from channel; channel then generates end-of-record interrupt signal

When reading cards in the single-word mode of transmission, a CHANNEL W ACTIVE TEST (CATW) should be issued before each WIM to ensure that the card reader has not become disconnected (read or feed check). Otherwise, the computer will "hang up" on the WIM should the buffer become disconnected before the desired number of columns has been read.

Controls and Indicators

The card reader control panel provides the following controls and indicators:

POWER ON	Pressing this switch causes the POWER ON and NOT READY indicators to be lighted.
NOT READY	This indicator is lighted whenever the card reader is in a not ready condition (and POWER ON has been pressed).
START	Pressing this switch (after POWER ON has been processed) puts the reader in a ready condition (turns off the NOT READY indicator).
EOF ON	If this switch is on (lighted and the card hopper is empty), the end-of-file condition is satisfied. If the switch is off (not lighted), the end-of-file condition is inhibited – whether the hopper is empty or not.
END OF FILE	This indicator turns on lights whenever the end-of-file condition is satisfied.
FEED CHECK	This indicator turns on whenever an improper feed cycle occurs.
READ CHECK	This indicator turns on whenever a malfunction occurs in the read station during a read cycle.
VALIDITY CHECK	This indicator turns on whenever an invalid character is read during a Hollerith read operation.
RESET	This switch is used to clear (turn off) the FEED CHECK, READ CHECK, and VALIDITY CHECK indicators.
STOP	Pressing this switch causes a not ready condition, turns on the NOT READY indicator, and stops the card reader after the card currently being read.
POWER OFF	Pressing this switch removes power from the card reader and turns off all indicators, except for EOF ON and END OF FILE.

CARD PUNCH

Before initiation of a card punch operation, the card punch should be loaded and tested as follows:

1. Loading procedure:
   a. Turn the card punch POWER switch ON.
   b. Load the hopper with blank cards.
   c. Press the START pushbutton on the card punch control panel. (This procedure initializes the coupler and establishes the ready condition for feeding and punching the cards.)
2. Testing procedure:
   a. Test channel (channel active test).
   b. Test card punch (card punch ready test).

Card Punch Tests

The card punch tests to follow are coded for channel W, using unit 1.

PUNCH BUFFER TEST　　SKS 12046B
(Skip if punch buffer empty)

This instruction is used to test the status of the punch buffer. If the punch buffer is clear (empty) and ready for loading when the punch buffer test is executed, the computer skips the next instruction in sequence and executes the following instruction. If the punch buffer is not clear when the punch buffer test is executed, the computer executes the next instruction in sequence (does not skip). The punch buffer is always clear if the punch is ready to feed and punch.

Card output example:

This program punches one card in Hollerith mode. It is written as a closed subroutine that uses channel W with interlace, card punch unit 1, and the end-of-record interrupt level. Index register X3 is used to count the 12 times the card image is presented to the punch.

Location	Instruction	Address	Comments
PCD	ZRO		This assembler directive saves the location for the subroutine entry.
	CLA STA	 SWICH	This pair of instructions clears a switch to be used later.
	LDA STA	PCD ENTR2	This pair of instructions places the main program mark address in location ENTR2.
	MIN	ENTR2	This instruction adds one to the stored contents of location ENTR2.
MCRDS	LDX	ROWS	This LDX instruction initializes the index register with 77777765B, which is -11 decimal.
	SKS	14046B	This instruction tests the card punch for a ready condition. The card punch is number 1 on channel W.
	BRU	*-1	This instruction is executed if the punch is not ready. It branches back to the punch ready test. 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.
GETRW	EOM	42646B	This instruction alerts the channel W interlace, connects card punch 1 to channel W and starts a card moving toward the punch station. Four characters per word and Hollerith format are specified.
	EOM	16000B	This instruction specifies the extended mode, arms the end-of-record interrupt level, and specifies output terminal function IORD.
	POT	PNCH	This instruction transmits the word count and starting address to the channel.
	BRU*	ENTR2	This instruction branches back to the main program.
PNCH	DATA	1202000B	The word in PNCH specifies that 20 words will be output from memory beginning in location 2000B. Note that the card image must be sent to the channel 12 times to punch a card.
ROWS	DATA	-11	

The main program is processed while the output is being performed by the channel. When finished with the output, an interrupt signal will be transmitted to interrupt level 33B, the end-of-record interrupt location for channel W.

Location	Instruction	Address	Comments
33B	BRM	ENTR2	This instruction branches and marks to location ENTR2.
⋮			
ENTR2	ZRO		This assembler directive saves a location for routine entry.
	BRX	GETRW	This instruction adds 1 to the base in the index. If the base has not been incremented to zero, the next instruction executed is at location GETRW. When the base is incremented to zero, the next instruction in sequence is executed. This index counts "row times" on each card.
	MIN	SWICH	This instruction sets a switch to indicate to the main program that the punch operation is complete.
	BRI	ENTR2	This instruction returns control to the main program and clears interrupt level 33B.

CARD PUNCH READY TEST SKS 14046B
(Skip if card punch ready)

The card punch is ready to feed and punch a card when all of the following conditions exist:

1. Card punch POWER switch is ON
2. Hopper is empty
3. Stacker is not full
4. Chip box is not full
5. Feed mechanism is operating properly
6. Card punch START pushbutton has been pressed
7. No feed or punch cycle is in process

The card punch is ready when the card punch ready test is executed, the computer skips the next instruction in sequence and executes the following instruction. If the card punch is not ready, the computer executes the next instruction in sequence (does not skip). This ready test should be made before each EOM instruction that initiates a punch cycle.

Card Punch Instructions

If the card punch is ready when the punch card EOM is executed, the punch punches one 80-digit row in a card (starting with row 12) and then waits for a new EOM. Since the card punch operates by rows, the card punch program must present the entire card image to the coupler 12 times for each card. The coupler examines the card image, and loads the punch buffer with the appropriate row image before each row is punched. After each row is punched, the punch buffer is cleared and the coupler waits for the next EOM. The card punch instructions to follow are coded without interlace, using channel W at 4 characters/word, for unit 1.

PUNCH CARD IN DECIMAL (Hollerith) EOM 2646B

This instruction starts the punch, causes a card to feed past the punch station, and specifies the Hollerith format. A transmission of 80 characters (20 words at 4 characters/word) must follow this instruction. The EOM and transmission of characters must be executed 12 times for each card to be punched.

PUNCH CARD IN BINARY EOM 3646B

This instruction is identical to PUNCH CARD IN DECIMAL except that the binary format is specified.

The EOM must be followed each time by a transmission of 160 characters (40 words at 4 characters/word). When the single-word mode of transmission is used for punching a card, each character transmission for a row must be followed by a TERMINATE OUTPUT ON CHANNEL W (TOPW) instruction. TOPW is automatically generated with interlaced outputs.

Error Conditions

If the card punch has been instructed to feed and punch a card and the card does not feed properly (or the punch buffer is not loaded at punch time), the error flip-flop in the channel is set.

LINE PRINTER OUTPUT

XDS 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 10, for XDS line printers consist of eight switches and indicators.

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.

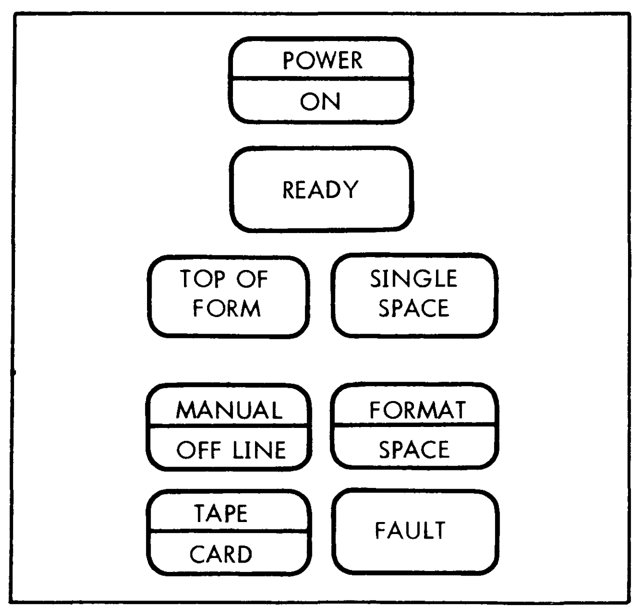


Figure 10. Printer Control Indicator Lights and Switches

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 printer 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 of a switch and two independent indicators. The program or the operator may initiate off-line operation, which is indicated by the illumination of the bottom half of this switch (OFF LINE). If the operator presses this switch to initiate off line operation, the top half (MANUAL) is also lighted, and remains lighted until the operator presses the switch again. OFF LINE is normally reset when the end-of-file is detected from the input unit. Pressing READY (when READY is lighted) also resets OFF LINE, 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.

[†]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.

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 controls single spacing. When printing with no format loop inserted in the printer, single upspacing occurs regardless of the channel specified.

LINE PRINTER TESTS

The line printer tests to follow are coded for channel W, using unit 1.

PRINTER READY TEST SKS 12060B
(Skip if printer ready)

This instruction tests the printer for a "ready" condition. The criteria for a printer "ready" condition are:

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

If the printer is ready when the printer ready test is executed, the computer skips the next instruction in sequence and executes the following instruction. If the printer is not ready, the computer executes the next instruction in sequence (does not skip). Since the printer tests ready while ejecting paper, the program should allow a definite time interval to pass (see PRINTER UPSPACE) after an upspace or skip to format channel instruction before executing a new upspace or skip to format channel. A dummy print instruction may be issued between two space instructions, which will provide the timing required. A printer ready test may be used to determine when the second paper space instruction may be executed.

PRINTER FAULT TEST SKS 11060B
(Skip if no printer fault)

This test determines if the printer has detected a parity error during a transfer of information from the printer buffer to the print hammers. If such an error occurs, a fault detector is set and the FAULT indicator is lighted. If the fault detector is set when the printer fault test is executed, the computer executes the next instruction in sequence (does not skip). If the fault detector is not set, the computer skips to the next instruction in sequence and executes the following instruction.

END OF PAGE TEST SKS 14060B
(Skip if not end of page)

This instruction tests the printer for paper position. If the paper is positioned at the end of page (specified by format channel 7), the computer executes the next instruction in se uence (does not skip). If the paper is not positioned at the specified end of page, the computer skips the next instruction in sequence and executes the following instruction.

Printer output example:

This program positions the paper at the top of the page and prints two lines with a single upspace between them. It is assumed that the printer is ready to print (or is becoming ready) after a print operation. This program, written as a closed subroutine, uses channel W with interlace, line printer 1, the zero-word-count interrupt level, and the end-of-record interrupt.

Location	Instruction	Address	Comments
PLP	ZRO		This assembler reserves a location for subroutine entry.
	CLA		This pair of instructions initializes a location, SWICH, which is later used to indicate that printing is completed.
	STA	SWICH	
	SKS	12060B	This instruction tests for printer ready.
	BRU	*-1	This instruction returns to the ready test; if the printer is not ready, the computer executes this instruction.
	EOM	11460B	This instruction causes the printer to move paper to the top of the page.
	EOM	42660B	This instruction alerts the channel W interlace, connects printer 1 to channel W, and specifies the 4 characters/word transfer mode.
	EOM	16200B	This instruction specifies the extended mode, arms the end-of-record interrupt level, and specifies output terminal function IOSD.
	POT	PRINT1	This instruction transmits the word count and starting address.
	BRR	PLP	This instruction branches back to the main program while the line is being printed.
PRINT1	DATA	2043720B	The word in PRINT1 specifies that 33 words will be output from memory beginning in location 3720B.

The main program continues while the data transfer and printing is being completed. When printing of the first line is completed, the end-of-record interrupt signal is transmitted to interrupt level 33B.

Location	Instruction	Address	Comments
33B	BRM	UPSPC	This instruction branches and marks to location UPSPC elsewhere in memory.
⋮			
UPSPC	ZRO		This assembler directive reserves a location for an entry.
	EOM	11660B	This instruction causes the printer to upspace one line.
	EOM	42660B	This instruction alerts the channel W interlace, connects printer 1 to channel W, and specifies the 4 characters/word transfer mode.
	EOM	15000B	This instruction specifies the extended mode, arms the zero-word-count interrupt level, and specifies output terminal function IORD.
	POT	PRINT2	This instruction transmits the word count and starting address to the channel.
	BRI	UPSPC	This instruction clears interrupt level 33B and branches back to the main program to await completion of the data transfer.
PRINT2	DATA	2043761B	The word in PRINT2 specifies that 33 words will be read from memory beginning in location 3761B.

The main program continues while the data transfer is being completed. When the word count equals zero during the transmission, the channel transmits the zero-word-count interrupt signal (to level 31B) and then disconnects from the printer after the last character has been transmitted.

Location	Instruction	Address	Comments
31B	BRM	DONE	This instruction branches and marks to location DONE elsewhere in memory.
⋮			
DONE	ZRO		This assembler directive reserves a location for an entry.
	MIN	SWICH	This instruction sets the printing-complete flag.
	BRI	DONE	This instruction branches back to the main program and clears interrupt level 31B. This is the final exit.

LINE PRINTER INSTRUCTIONS

The following line printer instructions are coded for channel W, using unit 1:

PRINT ON LINE PRINTER EOM 2660B

This instruction connects the line printer to channel W and specifies a character transmission of 4 characters per word.

This instruction is followed by the transmission of up to 132 characters. If the character count is less than 132, the characters are printed left-justified on the page. If the character count is more than 132, the printer produces an undetectable error. The printer disconnects from the channel after the line is printed; the channel then transmits a signal to the end-of-record interrupt level.

PRINTER OFF-LINE EOM 10260

This instruction places the printer off-line and initiates an off-line print operation. The selected input device (card reader 1 or magnetic tape unit 7) also goes off-line. (See "Off-Line Printing".)

PRINTER SKIP TO FORMAT CHANNEL EOM 1n460B

This instruction causes the printer to eject paper until the paper tape format loop detects the first punched hole in the channel specified by the number n (0 to 7). (See below for timing.)

PRINT UPSPACE n LINES EOM 1n660B

This instruction causes the printer to upspace n (0 to 7) lines. Consecutive upspacing instructions must be separated by a sufficient time delay. Otherwise, the two upspace instructions may be merged by the printer.

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

Upspace 1 line: 25 milliseconds (14,275 cycles)

Upspace more than 1 line: add 10 milliseconds (5690 cycles) for each additional line.

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 on channel W (TOPW) instruction. TOPW 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. Channel error – parity error or data overrun (character rate error) during transfer of information through buffer.
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 printer and the channel may still be used by the computer for other input/output operations (e.g., card reading on card reader 2, card punch, paper tape reader/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 5). 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.

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 7.)
2. Place paper at top of form, as desired, by means of the TOP OF FORM switch.
3. Select desired input device (magnetic tape 7 or card reader 1) 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.

Table 5 . 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 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 instruction to print off-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, 17B.

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.

MAGNETIC TAPE INPUT/OUTPUT

MAGNETIC TAPE FORMAT

All magnetic tape units used by the XDS 940 System are IBM-compatible. Tape reels can contain up to 2400 feet of tape. A reflective marker is placed on the back of the tape, approximately 10 feet from its beginning, to indicate the load point. The leading 10 feet are used 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 placed 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 10 feet of leader and enough tape to hold a record of 9600 characters at 200 bpi density after the end-of-reel marker is sensed.

Characters are recorded on tape in seven parallel tracks. Six of the tracks are used for information; the seventh track is a parity check. Both even and odd parity are used. Data is recorded in the binary mode using odd parity. In this mode the 6-bit characters from the channel are recorded without alteration. Data is recorded in the binary-coded decimal (BCD) mode using even parity. In this mode, characters from the channel are transformed to IBM standard BCD interchange code (see Appendix A).

Information on tape is arranged in blocks that may contain one or more records. A record may be any length within the capacity of available core storage in the computer. Records or blocks of records are separated on tape by a record gap (section of blank tape) about 3/4 inch long. In writing, the gap is automatically produced at the end of a record or block. Reading begins with the first character sensed after the gap and continues until the next gap is encountered.

An interrecord gap, followed by a special, single-character record, is used to mark the end of a file of information. The character is a tape mark (17B) and is recorded by writing a one-word record in BCD with 1-character/word format. One or more files may be written 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 will consider any record that contains only tape mark (17B) characters as an end-of-file. All such characters will be read into memory as requested.

As information is written, an odd-even count is made of the number of 1-bits in each channel. At the end of each record a bit is written for each channel so that the total number of 1-bits in each track will be even. This check is always even whether the character parity is even or odd. The character containing these check bits is called the longitudinal parity character and is written slightly past the end of record information in the block.

Since the longitudinal check character always reflects an even parity check for each channel, in the BCD mode, the check character itself will always have 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 channel even though the record itself is correct. As a general rule, the error indicator should be ignored after a reverse scan operation.

It is possible to write tape in a 1-, 2-, or 3-characters/word mode provided characters can be supplied at a sufficient rate. On reading, however, the tape unit uses the character count to ascertain when it has read two characters and can look for gap. If a 1-character/word read were started, a single noise character would stop the tape. In reverse scan, a 1-character/word operation would cause the tape to stop after detecting the longitudinal check character at the end of the record with the tape positioned in the area of recorded information.

As a general rule, tape units should be programmed for 3 or 4 characters/word if possible. The write-tape-mark operation is an exception to this rule. All reverse scan operations must be in the 4-characters/word mode.

The tape ready test should be used between tape operations of opposite direction to ensure that the tape unit stops and reverses. It is advisable to terminate tape writing by erasing several inches of tape whenever subsequent resumption of recording is anticipated. This will eliminate the effects of a possible extraneous character which might arise through subsequent tape repositioning.

MAGNETIC TAPE UNIT TESTS

The magnetic tape unit tests to follow are coded for channel W, with n being the number (0-7) of the magnetic tape unit.

TAPE READY TEST — SKS 1041nB
(Skip if tape not ready)

Tape unit n is tested for not ready. If the tape is not ready, the next instruction in sequence is skipped and the following instruction is executed. If the tape is ready, the instruction in sequence is executed. A tape is not ready if:

1. There is no physical unit set to the logical number being tested,
2. The selected unit is not in the automatic mode, or
3. The selected unit is in motion for any operation.

FILE PROTECT TEST — SKS 1401nB
(Skip if tape not file protected)

Tape unit n is tested for file protecting. If the file protect ring is inserted, the next instruction in sequence is skipped and the following instruction is executed. If not inserted, the next instruction in sequence is executed. The skip will not occur if there is no logical unit n on the channel. This instruction should be used before any write operation to determine whether it is possible to perform the write operation.

BEGINNING OF TAPE TEST — SKS 1201nB
(Skip if not beginning of tape)

Tape unit n is tested at the beginning of the tape. If it is not positioned on the load-point marker, the next instruction in sequence is skipped and the following instruction is executed. If positioned at the load-point marker, the next instruction in sequence is executed. The skip will not occur if there is no logical unit n on the channel.

END OF TAPE TEST — SKS 1101nB
(Skip if not end of tape)

Tape unit n is tested to see if it has sensed the end of the tape. If the tape unit has not sensed the end-of-reel marker, the next instruction in sequence is skipped and the following instruction is executed. If the end-of-reel marker has been sensed, the next instruction in sequence is executed. The end-of-reel condition is reset when the tape is moved backwards over the end-of-reel marker. The skip will not occur if there is no logical unit n on the channel.

DENSITY TEST, 200 BPI[†] — SKS 1621nB
(Skip if not 200 BPI)

Tape unit n is tested for being set at 200 bpi density. If not, the next instruction in sequence is skipped and the following instruction is executed. If so, the next instruction in sequence is executed.

DENSITY TEST, 556 BPI[†] — SKS 1661nB
(Skip if not 556 BPI)

Tape unit n is tested for being set at 556 bpi density. If not, the next instruction in sequence is skipped and the following instruction is executed. If so, the next instruction in sequence is executed.

DENSITY TEST, 800 BPI[†] — SKS 1721nB
(Skip if not 800 BPI)

Tape unit n is tested for being set at 800 bpi density. If not, the next instruction in sequence is skipped and the following instruction is executed. If so, the next instruction in sequence is executed.

TAPE END-OF-FILE TEST[†] — SKS 13610B
(Skip if not end-of-file)

The tape control unit is tested to determine if a tape under its control encountered an end-of-file during the last read or scan operation. If not, the next instruction in sequence is skipped and the following instruction is executed. If end-of-file was encountered, the next instruction in sequence is executed. The end-of-file indicator remains set until another tape operation is called for.

[†]Note: These instructions apply only to 41.7-kc and 96-kc magnetic tape systems.

Magnetic tape input example:

This program reads one record from magnetic tape unit 1 on channel W with interlace. The program is written as a subroutine that uses the end-of-record interrupt level. It is assumed that the tape is not at the beginning or the end of tape.

Location	Instruction	Address	Comments
RMT	ZRO		This assembler directive saves a location for the subroutine entry.
	SKS	10411B	This instruction tests (for ready) magnetic tape 1, channel W. If magnetic tape 1 is ready to perform an input/output operation, the next instruction in sequence is executed; if not, the next instruction is skipped and the following one is executed.
	BRU	*+2	This instruction skips one instruction.
	BRU	*-2	This instruction branches back to the tape ready test. An exit to a routine that determines reasons for the nonready condition can be placed here.
	EOM	42611B	This instruction alerts the channel W interlace, connects channel W to magnetic tape 1, and starts tape motion. The 4 characters/word and BCD format are also specified.
	EOM	16000B	This instruction specifies the extended mode, arms the end-of-record interrupt level, and specifies input terminal function IORD.
	POT	REDTP	This instruction transmits the word count and starting address to the channel.
	BRR	RMT	This instruction branches back to the main program.
REDTP	DATA	6202000B	The word in REDTP specifies that one record, or 100 words, whichever is smaller, will be read into memory beginning in location 2000B. Any remaining words in the record after the first 100 will be ignored.

The main program is processed while the input operation is being performed by the channel. When finished, the end-of-record interrupt signal will be transmitted to location 33B.

Location	Instruction	Address	Comments
33B	BRM	COMPL	This instruction branches and marks to COMPL to finish the read operation.
:			
COMPL	ZRO		This assembler directive saves a location for the routine entry.
	CETW		This instruction tests for error in channel W. If an error is detected, the next instruction in sequence is executed; if not, the next one is skipped and the following instruction is executed. The octal configuration of this instruction is 0 40 11000.
	BRM	ERTST	This instruction branches to an assumed routine that repeats the read operation a few times and, if the error continues, informs the operator.
	BRI	COMPL	This instruction returns control to the main program and clears interrupt level 33B.

TAPE GAP TEST[†] SKS 12610B
(Skip if tape not in gap)

The tape control unit is tested to see if a tape under its control is in motion in the gap following a record; if not, the computer skips the next instruction in sequence and executes the following instruction; if so, the computer executes the next instruction in sequence.

When the tape unit detects the gap at the end of a record and has checked the longitudinal parity character, it generates the gap signal. This signal remains true for approximately 1 millisecond (approximately 570 computer cycles). During this time, the test instruction does not skip, and the tape may be given a command to continue in the direction it is going. If so programmed, the tape continues without stopping. If the record encountered should be an end-of-file, the gap signal does not become true, the tape always stops, and the test instruction skips.

MAGPAK TEST SKS 1021nB
(Skip if not MAGPAK)

Tape unit n is tested for being a MAGPAK. If the tape unit is not a MAGPAK, the computer skips the next instruction

[†] This instruction applies only to 41.7-kc and 96-kc magnetic tape systems.

in sequence and executes the following instruction. If the tape unit is a MAGPAK, the computer executes the next instruction in sequence.

READING FROM MAGNETIC TAPE

Once a tape is started with a read binary or read BCD EOM/EOD, it continues until an end-of-record gap is detected. If the computer does not instruct it to continue, it will then stop in the middle of that gap. When the tape stops, the tape unit disconnects from the channel. If an end-of-file is encountered, the tape control unit sets its EOF indicator. This indicator can be tested by the central processor and will remain set until a new EOM/EOD is given to a tape unit on that channel. The tape always stops after the tape mark.

The EOF character (001111 in binary) is read into memory along with its check character. In a 4-character/word read, this appears in the first word of the input area as a 17170000B word.

Once a record has been written on tape, it cannot be certain that any subsequent record previously written can be read. This means that a record in the middle of a file cannot be updated or rewritten if following records need to be read.

Any error detected, either by the channel (in the character parity) or by the control unit (with longitudinal parity), sets the error indicator in the channel. When an error is detected in reading, the tape should be backspaced over the erroneous record and a reread attempted.

If the end-of-reel marker is encountered while reading, the end-of-reel indicator in the tape unit is set and may be interrogated by the program at any time. An end-of-file is normally used to indicate the end-of-record information on tape. It is possible, however, to use the end-of-reel indicator to mark the last record on the reel.

Backspace

A backspace record is implemented using the scan feature. A scan reverse EOM/EOD is used to start the tape in reverse. The channel is then loaded with a 00 function (IORD) with a 0 count, and the end-of-record interrupt level armed. When the channel signals that the operation is complete, the tape is situated with the read-write head in front of the last record scanned.

Scan

A scan operation is similar to a read operation except that the channel shifts the characters through its word assembly register, but does not consider a word complete until a tape gap is encountered. When the gap is reached, the channel uses the last four characters in the word assembly as the only word read from the record. When scanning in reverse, the word consists of the last four characters scanned, which are the first four logical characters of the record. These characters will be assembled in reverse. For example, if the first four characters of the record were ABCD and the record was scanned in reverse, these would appear as DCBA in the word stored for that record.

The scan is useful for reverse searching on the first word of the records in the file being searched. In this case, the tape is started in a reverse scan and the channel interlace loaded with terminal function 10 (IORP), with a word count of 1, and the zero word count interrupt level armed. When the beginning of the record is reached, the channel interrupts the program with the zero word count interrupt level. The program checks the first word in the record against a search key. If they agree, then the program need only wait for the channel to become inactive and the record may be read forward. If the record is not the desired one, the program reloads the channel interlace to scan the next record and gives another scan reverse without waiting for the channel to become inactive.

Magnetic Tape Read/Scan Instructions

The magnetic tape read instructions to follow are coded for channel W without interlace, using tape unit n in the 4-characters/word mode.

READ TAPE IN BINARY — EOM 361nB

Tape unit n is started in a binary read mode.

READ TAPE IN DECIMAL (BCD) — EOM 261nB

Tape unit n is started in a BCD read mode.

SCAN FORWARD IN BINARY — EOM 363nB

Tape unit n is started forward in a binary scan mode.

SCAN FORWARD IN DECIMAL (BCD) — EOM 263nB

Tape unit n is started forward in a BCD scan mode.

SCAN REVERSE IN BINARY — EOM 763nB

Tape unit n is started in reverse in a binary scan mode.

SCAN REVERSE IN DECIMAL (BCD) — EOM 663nB

Tape unit n is started in reverse in a BCD scan mode.

MAGNETIC TAPE UNIT CONTROLS

The following instructions are used for control of magnetic tape units. These instructions are EOMs in the input/output control mode.

REWIND — EOM 1401nB

Tape unit n is started in a rewind. Once started, the tape continues in rewind until the load point marker is sensed; it then stops, and after 1 second (to allow the drive capstans to return to normal speed) generates a ready signal.

CONVERT READ TO SCAN — EOM 14000B

The tape unit currently in a read mode on the channel is instructed to convert from the read mode of operation to the scan mode of operation.

SKIP REMAINDER OF RECORD[t] EOM 13618B

The tape unit currently on the channel is instructed to skip the remainder of the record being read.

WRITING ON MAGNETIC TAPE

Once a tape unit is ready and the file protect ring is on the tape reel, that is, the file protect test is false, a write operation can be initiated. The tape will start and remain in motion until the termination signal from the buffer is received. The tape control unit will then write the remaining characters of the record and the longitudinal check character. When the check character is read by the read-after-write head, the tape will signal the channel that gap has been reached. If no further write instruction is received within 1 millisecond, the tape is stopped and disconnected.

An end-of-file character should be written (or a segment of tape erased) after a series of records have been written, if the user wishes to backspace or rewind and then expects to return at some later time to record additional information at the end of the previous series of records. This practice provides positive identification of the end of a record and facilitates return to a specific location on the tape. If this method is not used, there is a possibility that the tape will not subsequently stop in the same location at the end of the series of records as it did when the last record was written. This would leave a segment of tape in the gap which has not been written and may cause erroneous operation when the tape is read.

---

[t]Note: This instruction applies only to 41.7-kc and 96-kc magnetic tape systems.

Erase Procedures

In addition to writing under program control, magnetic tape can also be erased under program control. Tape may be erased by addressing it with an erase unit address. When a tape is addressed with an erase unit address, it operates as though it were in a write mode, except that no information is recorded. The program or interlace supplies the count of the number of words to be erased.

This type of erase is useful for the correction of a write error. When a write error occurs, an ERASE TAPE IN REVERSE is given to start the tape in reverse. Then the same count, used to write the record originally, is loaded to control the erase. This procedure ensures that the tape always returns to the beginning of the erroneouse record, even if a bad spot on the tape might appear as a gap. The record may now be rewritten. If the write still produces an error, the record is erased backwards and then an erase forward, using the same count, bypasses the section of tape where the difficulty occurred. The record may now be rewritten on a new section of tape.

Long Gap. The erase procedure is used to produce 3.75 inches of blank tape between the load point and the first record and between a record and the end-of-file mark. This is accomplished by erasing 150 words at 200 bpi density, 417 words at 556 bpi density, or 600 words at 800 bpi density.

End of File. Writing an end-of-file record is accomplished by the following sequence:

1. Erase a long gap.
2. Load the channel interlace with a word count of 1 and the address of a word containing the value 17000000B.
3. Issue a 1-character/word, BCD write tape instruction.

Magnetic tape gather-write example:

This program writes one record on magnetic tape. The data written in the record are gathered from three non-contiguous areas of memory. The program is written as a closed subroutine that uses the zero-word-count interrupt, magnetic tape 1, and channel W with interlace.

This program is written to clarify programming for magnetic tapes. Extra programming is not included to save the contents of the A register or the index register for the main program.

A scatter-read operation can be performed with an almost identical program. The difference is the exchange of the read instruction with the write instruction and the deletion of the file-protect test.

Location	Instruction	Address	Comments
GWMT	ZRO		This assembler directive saves a location for the subroutine entry.
	CLA		This pair of instructions clears location COUNT for use later as a switch.
	STA	COUNT	
	SKS	10411B	This instruction tests magnetic tape 1 on channel W for being ready.
	BRU	*+2	This instruction branches two locations ahead. This instruction is executed if the magnetic tape is ready.
	BRU	*-2	This instruction branches back to the tape ready test.

(continued)

Location	Instruction	Address	Comments
	SKS	14011B	This instruction tests whether the file protect ring is present on the tape reel. If so, the next instruction is skipped and the following one is executed.
	BRM	OPER	This instruction branches and marks to an assumed routine to call the operator and instruct him to insert file protect ring on magnetic tape 1.
	LDA STA MIN	GWMT FAST FAST	These three instructions place the marked subroutine entry location plus one into location FAST.
	BRU	FAST+1	This instruction branches around location FAST.
FAST	ZRO		This assembler directive saves a location for entry to the multiple write area of the subroutine.
	LDX	COUNT	This instruction loads the index register with the contents of location COUNT. This is used in picking up the proper input/output control instructions.
	LDA SKG	OKAY COUNT	These two instructions determine when the write operation is complete.
	BRI	FAST	This instruction is executed if the write operation is complete. When it is, location COUNT contains the value 6 and the active interrupt level 31B is cleared.
	EOM	42651B	This instruction (executed if the write operation is not complete) alerts the interlace in channel W for subsequent loading, connects magnetic tape 1 to channel W, specifies BCD transfer mode, and starts the tape moving. The 4 characters/word mode is specified.
	EXU	A, 2	This instruction executes the EOM located in address A modified by the contents of the index register. This process is repeated for the output words in A+2 and in A+4; then, the test in location FAST+3 causes a final branch back to the main program.
	POT	A+1, 2	This instruction transmits the word count and starting address to the channel.
	MIN MIN	COUNT COUNT	This pair of instructions adds 2 to the contents of location COUNT.
	BRI	FAST	This instruction branches back to the main program.
A	EOM	15600B	This EOM specifies output function 11 (IOSP) and arms the zero-word-count interrupt level.
A+1	DATA	6202000B	The word in location A+1 specifies that 100 words will be output from memory beginning in location 2000B.
A+2	EOM	15600B	This EOM specifies output function 11 (IOSP) and arms the zero-word-count interrupt level.
A+3	DATA	14402500B	The word in location A+3 specifies 200 words from memory beginning in location 2500B.
A+4	EOM	15000B	This EOM specifies output function 00 (IORD) and arms the zero-word-count interrupt level.
A+5	DATA	6203000B	The word in location A+5 specifies 100 words for memory beginning in location 3000B. Upon completion of the output of this subrecord, the channel disconnects.
OKAY	DATA	6	This is the value used in the completion tests.

The main program is processed while the output is being performed by the channel. When output is completed, the channel transmits the zero-word-count signal to interrupt level 31B.

31B	BRM	FAST	This instruction branches and marks to location FAST.

Magnetic Tape Write Erase Instructions

The magnetic tape write instructions to follow are coded for channel W without interlace, using tape unit n in the 4-characters/word mode. EOM or EOD instructions to the tape units specify start-without-leader since the tape unit generates gap on all write operations automatically. Thus, it is not necessary for the starting EOM to call 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.

WRITE TAPE IN BINARY EOM 365nB

Tape unit n is started in a binary write mode.

WRITE TAPE IN DECIMAL (BCD) EOM 265nB

Tape unit n is started in a BCD write mode.

ERASE TAPE FORWARD EOM 367nB

Tape unit n is started in an erase mode.

ERASE TAPE IN REVERSE EOM 767nB

Tape unit n is started in reverse in an erase mode.

RAPID-ACCESS DATA (RAD) FILE

The XDS RAD file consists of a controller and one to four devices, with each device containing either one or two logical storage units. The logical storage units are available in two sizes: 524,288 characters (6 bits per character) or 1,048,576 characters. Thus, the maximum capacity per RAD controller is 8,388,608 characters.

Each logical storage unit contains 64 bands (32 bands for a 524,288-character unit), each band contains 64 sectors, and each sector contains 256 characters (64 words). Thus, each band contains 16,384 characters (4096 words). Areas of the RAD storage are thus addressed by specifying the particular unit, band, and sector. Each particular band is read/recorded with a separate set of read/write heads, thereby eliminating the mechanical complexity and positioning time characteristic of movable arm disc files.

Due to the data organization of the bands, reading/recording of 6-bit characters must be done in multiples of two characters (12 bits), which is what the controller requires. This means that to the controller, a 24-bit word appears as two 12-bit characters. This is the reason for selecting the 2 character/word mode of the EOD when programming the RAD file.

The minimum data unit that can be written on the RAD is a sector (64 words) and all transmission must begin on a sector boundary. If the interlace word count controlling a write operation is not an integer multiple of 64, the remainder of the last sector will be written as zeros (erased). The number of words transmitted to memory for a read operation is specified by the interlace word count.

Sector address incrementing is automatically performed by the controller when more than one sector is to be transferred. Unless inhibited by the ALERT RAD AND INHIBIT INCREMENT instruction, the band number will also be incremented after the last sector in each band (if transmission crosses a band boundary). Addressing is continuous only within a logical unit (not continuous from unit to unit). Thus, if a million characters of storage are set up as two .5-million-character logical units, it is not possible to increment automatically across the logical unit boundary.

FILE PROTECTION

RAD units contain a provision for manual write protection. A group of toggle switches selectively inhibit writing on band groups associated with each switch. There are a total of eight switches for each logical unit, with each switch controlling eight bands (32,768 words of storage). The write-protect switches can be tested, under program control, by executing the RAD FILE PROTECT TEST (see below).

RAD FILE TESTS

The RAD file tests to follow are coded for RAD file 1 on direct-access communication channel E.

RAD CONTROLLER READY TEST SKS 50026B
(Skip if controller ready)

If the RAD controller is ready when the ready test is executed, the computer skips the next instruction in sequence and executes the following instruction. If the RAD controller is not ready, the computer executes the next instruction in sequence (does not skip). The controller will respond with a not ready indication if:

1. The RAD file power is below safe read/write limits,
2. The controller has been addressed by a READ RAD or a WRITE RAD instruction, but the RAD controller has not yet begun to read/write, or
3. The read/write circuitry is currently in use.

This ready test should be made before each alert EOD instruction that initiates a read or a write operation.

RAD FILE PROTECT TEST SKS 53026B
(Skip if RAD file not protected)

The RAD file band addressed by the preceding POT instruction is tested for being write protected. If the address band is not write protected (i.e., the write-protect switch for the band is off), the computer skips the next instruction in sequence and executes the following instruction. If the addressed band is write protected, the computer executes the next instruction in sequence (does not skip). The controller responds with a protected indication when the write-protect switch for the addressed band is on, even if the band is unimplemented (in the case of a .5-million-character storage unit).

RAD CONTROLLER ERROR TEST SKS 51026B
(Skip if no RAD controller error)

If the RAD controller error indicator is not set, the computer skips the next instruction in sequence and executes the following instruction. The computer executes the next instruction in sequence (does not skip) if the RAD controller error indicator is set, as the result of one or more of the following conditions:

1. The controller has detected a device fault (malfunction).
2. A write operation has been attempted for a band that is write protected.

Note: For the above error conditions, the RAD controller automatically disconnects from the channel (clears the channel unit address register). Thus, the channel returns to the inactive condition. However, the interlace registers are not affected and they contain the word count and memory address that existed at the time of the error. Since the interlace word count is not decremented after the channel is disconnected, the zero-word-count signal is not transmitted to the interrupt system; however, the channel transmits the end-of-record interrupt signal (if the interrupt level is armed) when the channel is disconnected.

3. The RAD controller was not in a ready condition when the instruction ALERT RAD or the instruction ALERT RAD AND INHIBIT INCREMENT was executed. In this case, only the RAD controller error indicator is set (the channel is not disconnected and the current operation is not affected).

RAD FILE INSTRUCTIONS

The RAD file instructions to follow are coded for RAD file number 1 on direct-access communication channel E.

ALERT RAD EOD 10026B

This instruction alerts the RAD file to receive the word to be transmitted by the POT instruction that immediately follows. The word transmitted by the POT instruction is assumed to be in the following format:

000	Unit	Band	Sector
0–8	9–11	12–17	18–23

If the designated unit/band address is not existent in the RAD file, an erroneous operation occurs when data transfer is attempted. No error indicator is set for this condition.

The optimum time for execution of the POT instruction is when the current sector address is 1 less than the sector address in the word to be transmitted; otherwise, the computer must wait for the beginning of the addressed sector to be available for a read/write operation. The EOD to alert the RAD file should be preceded by a RAD controller ready test to insure that the RAD file is ready when the EOD is executed; otherwise, an error condition occurs.

ALERT RAD AND INHIBIT INCREMENT EOD 11026B

This instruction performs all the functions of the ALERT RAD instruction. In addition, this instruction inhibits the RAD controller from incrementing the current band address. This allows the program to begin a read/write operation in the middle of a band, read/write to the end of the band (sector 64), and continue the read/write operation at sector 0 of the same band. The restrictions that apply to ALERT RAD also apply to this instruction.

ALERT TO STORE SECTOR EOD 1n226B

This instruction alerts unit n (where $0 \leq n \leq 7$) of the RAD file to transmit its current sector address to the computer. This instruction must be followed immediately by a PIN instruction, which stores the sector address in the effective location as follows:

000000	Sector
0–17	18–23

READ RAD FILE EOD 2226B

This instruction connects the RAD file to the channel and initializes the channel to assemble 2 characters/word (12 bits per character).

Execution of the EOD instruction also resets the RAD controller error indicator (unless the RAD controller is not ready when the EOD is executed).

WRITE RAD FILE EOD 2266B

This instruction connects the RAD file to the channel and initializes the channel to transmit 2 characters/word (12 bits per character).

Execution of the EOD instruction also resets the RAD controller error indicator (unless the RAD controller is not ready when the EOD is executed).

ERROR CONDITIONS

RAD Controller Error Conditions

The conditions that cause the RAD controller to set its error indicator, which is independent from the channel error indicator, are listed in the description of the RAD controller error test. The RAD controller error indicator is reset whenever a READ RAD or WRITE RAD instruction is executed (unless the RAD file is not ready when the instruction is executed; in which case, the RAD controller error indicator is set).

Channel Error Conditions

For a write operation, the channel error indicator will only be set if a data overrun (rate error) exists. This condition occurs if the channel is not prepared to transmit the next character when requested by the controller.

For read operations, three distinct conditions will set the channel error.

1. A data overrun (rate error) occurs if the channel is not prepared to receive the next character from the coupler.
2. The channel error indicator is set if the channel detects a parity error in the input character.
3. The channel error indicator is set if the controller detects a parity error in the check character at the end of a sector.

The channel error indicator is reset whenever a READ RAD or a WRITE RAD instruction is executed.

Recovery Procedure

Disregarding critical applications, it is recommended that no more than three attempts be made to read or write a portion of the file.

PROGRAMMING CONSIDERATIONS

Immediate Access

The rotational latency (35 milliseconds maximum) of the RAD file can be minimized when large blocks of data are transferred between the computer and the RAD file. This type of large block transfer is frequently encountered in applications such as time sharing and very large simulation problems employing program "overlay" techniques.

A combination of hardware provisions and programming techniques is used to minimize rotational latency. The current sector address of each unit is continually maintained in the controller. As the bands revolve, successive sector addresses are noted and counted, as a 6-bit number in the range 00B through 77B. The current sector address of any unit can be read by the computer's program at any time by an EOD-PIN sequence. Thus, the program can compute the optimum starting point for a data transfer, as illustrated in the following example:

Assume the program is to transfer 10000B (4096) words from core memory location 5720B through 15717B onto sector locations 4200B through 4277B (band 42B, all sectors). Assume that the current sector address is 43B. These steps occur:

1. The program tests for controller ready with an SKS instruction.

2. The program executes an EOD-PIN sequence. The current sector address (43B) is thus read into the computer.

3. Two is added to the sector address, making it 45B. This insures that one sector time is available to the program before reading or writing occurs. (The unit could be very near the end of sector 43B when the sector address is read; hence, the unit would be into sector 44B before a read or write could be initiated – therefore imposing a full rotational delay.)

4. The program sets up two I/O file operations.

   a. The first causes core locations 12420B through 15717B to be written onto sectors 4245B through 4277B.

   b. The second causes core locations 5720B through 12417B to be written onto sectors 4200B through 4244B.

In this example, 4096 words are transferred in about 35 milliseconds. If normal programming had been used (with the entire record started at sector zero), the operation would have taken almost 50 milliseconds, due to the latency encountered while the unit rotated from sector 44 to sector 00. Thus, the immediate access capability of the RAD file reduced the transfer time by about 30 percent.

Use of the Channel Interrupt Levels. Care should be exercised when using the zero-word-count interrupt level and the channel zero count test, because the controller may disconnect (before zero count) for any of the reasons listed in the description of the RAD controller error test.

The occurrence of an end-of-record interrupt signal (or a skip as the result of a channel active test) is not necessarily synonymous with the controller becoming ready. If the interlace word count is modulo 64, the controller on output must (after zero count) write the last data characters and the check character before indicating that it is ready. If the interlace word count is not modulo 64, the coupler will be busy after the end-of-record signal is transmitted and until trailing zero characters and the check character are written (or until the end of the current sector is reached and the check character is verified for reading).

Use of the "Sector Start" Interrupt Level. As an optional feature, the RAD file controller can inform the program, via one of the special systems interrupt levels, that a specific unit is about to reach the beginning of a specific sector. The procedure for using this feature is outlined in the following steps.

1. The program (or I/O routine) enables the interrupt system. If the optional arming feature is present, the program must also arm the interrupt level assigned to this function. (The assignment of special systems interrupt levels is unique to each computer system, as determined by the customer.) The program must also load the interrupt location with the appropriate BRM instruction.

2. The program sets up the I/O operation, which includes executing the necessary ready tests, the alert controller EOD and subsequent POT, file protect test, and the EOD and POT instructions that alert and load the channel interlace. The read/write instruction is not executed at this time.

3. The program branches out of the I/O set-up and continues with its other processing.

4. When the addressed RAD unit has rotated to the point where 28 microseconds (approximately 16 computer cycles) remain before the addressed sector reaches the read/write heads, the RAD controller transmits an interrupt signal to the special systems interrupt level.

5. The BRM instruction in the interrupt location branches and marks to an interrupt-servicing routine that contains the read/write instruction. The interrupt-servicing routine then branches back to the main program (which may again be interrupted, via the channel zero-word-count or end-of-record interrupt levels, for error checking and for setting an operation-completed switch).

RAD output example

This program writes a record of 64 words (from core memory locations 1000B through 1077B) onto RAD file 1, storage unit 0, band 1, sectors 0 through 77B (64). The program is written as a subroutine that uses channel E without interrupts.

Location	Instruction	Address	Comments
WRAD	ZRO		This assembler directive saves a location for the subroutine entry.
	SKS	54000B	This instruction tests channel E for being active. If channel E is active, the computer executes the next instruction in sequence; otherwise, it skips the next instruction and executes the following instruction.
	BRU	*-1	This instruction branches back to the channel active test.
	SKS	50026B	This instruction tests RAD controller 1 for being ready. If the controller is not ready, the computer executes the next instruction in sequence; otherwise, the computer skips the next instruction and executes the following one.
	BRU	*-1	This instruction branches back to the controller ready test.
	EOD	50026B	This instruction alerts the RAD controller to receive the unit, band, and sector address for the data transfer.
	POT	ADDR	This instruction transmits the unit, band, and sector address to the RAD controller.
	SKS	53026B	This instruction tests the RAD controller to determine whether the band address just transmitted is write protected. If the band address is write protected, the computer executes the next instruction in sequence; otherwise, the computer skips the next instruction and executes the following one.
	BRM	PROTECT	This instruction branches and marks to an assumed routine to handle the error condition caused by an attempt to write on a protected band.
	EOD	50000B	This instruction alerts the channel E interlace.
	EOD	14200B	This instruction specifies output function 01 (IOSD) and disarms the zero-word-count and end-of-record interrupt levels.
	POT	OUTPUT	This instruction transmits the word count and starting address to the channel.
	EOD	02266B	This instruction connects the RAD controller to the channel, resets the RAD controller error indicator, specifies the forward, no leader, decimal, 2 characters/word operating mode, and starts the data transfer.
	SKS	50026B	This instruction tests the RAD controller forbeing ready. The controller becomes ready when the output operation is completed.
	BRU	*-1	This instruction branches back to the RAD controller ready test.
	SKS	51026B	This instruction tests for RAD controller error.
	BRU	ERROR	This instruction is executed if a RAD controller error condition has occurred.
	SKS	51000B	This instruction tests for channel E error.
	BRU	RETRY	This instruction branches to a routine that attempts to recover from the error.
	BRR	WRAD	This instruction returns to the main program.
ERROR	SKS	53026B	This instruction determines whether the error condition was caused by a device malfunction or by an attempt to write on a protected band.
	BRU	PROTECT	This instruction is executed if the error was because of an attempt to write on a protected band.
	BRU	OPER	This instruction branches to an assumed routine that notifies the operator of the device malfunction.
	BRR	WRAD	This instruction returns control to the main program.
ADDR	DATA	100B	The word in location ADDR designates unit 0, band 1, sector 0 of the RAD file.
OUTPUT	DATA	4001000B	The word in location OUTPUT specifies a count of 64 words and a starting address of 1000B.

ASYNCHRONOUS COMMUNICATIONS INTERFACE EQUIPMENT

The asynchronous communications interface equipment (CTE) permits the transfer of 11-unit, 10-character/second teletype information between the computer and up to 64 Data-Phone data sets. The CTE-10 controller interfaces between the computer POT and PIN connectors and up to 16 CTE-11 interface units. Each CTE-11 interfaces with four Data-Phone data sets. The system is expandable by the addition of a CTE-12 controller extender, which allows the addition of 16 more CTE-11 units (for a total of 128 channels).

The interface equipment uses four interrupt levels. An optional end-of-message detector uses a fifth interrupt level. Each interface channel has a transmit and a receive character buffer, which perform all necessary serial-to-parallel and parallel-to-serial operations and provide the necessary control timing. The transmit buffers are self-contained, resulting in true asynchronous operation, and each transmit buffer shifts independently of the others. Flags are associated with each transmit and receive buffer to indicate when a character has been transmitted or received. Two additional flags are used to indicate a change in status of the carrier-detect signal from the Data-Phone data set. The flags are continuously scanned by a scanning unit within the interface controller. Upon encountering a "raised" flag, the scanner stops and issues an interrupt signal unique to the flag. At this time the scanner register contains the 7-bit channel address for the raised flag. The scanner is subsequently restarted when the computer reads this address by an EOM-PIN sequence.

A program option is provided to suppress scanning of a particular transmit buffer flag, thus prohibiting an interrupt at the completion of the transmit operation. Also, a special configuration of the SKS instruction is provided for testing for transmit buffer ready, for data set ready, and for carrier present. Any Data-Phone data set can be activated or deactivated under computer control.

CTE INSTRUCTIONS

SELECT CTE INTERFACE EOM 77777B, 2

This EOM instruction selects the asynchronous communications interface; that is, it causes the interface controller to be connected to the direct parallel input/output lines of the computer. The interface remains connected until a POT or PIN instruction is executed.

OUTPUT CHARACTER AND SET CTE INTERRUPT CONTROL

When a POT instruction follows the select interface EOM, the effective word of the POT instruction is transmitted to the interface controller. The effective word is assumed to have the following format

Character	NT	I	D	(unused)	Channel
0–7	8	9	10	11–16	17–23

Bit(s)	Function
0-7	The character to be transmitted on the channel specified by bits 17-23.
8(NT)	If this bit is a 1, the character in bit position 0-7 is not to be transferred to the transmit buffer; if this bit is a 0, the character is to be transferred to the transmit buffer.
9(I)	If this bit is a 1, a transmit interrupt signal is issued when the transmission is completed; if this bit is a 0, no such interrupt signal is issued.
10(D)	If this bit is a 1, the data set is to be deactivated; if this bit is a 0, the data set is to remain (or become) active.
11-16	Ignored
17-23	The address of the channel on which the character is to be transmitted.

READ CTE CHANNEL ADDRESS AND DATA

When a PIN instruction follows the select interface EOM, the effective location contains the following information after the PIN instruction is executed:

Character	DO	00000000	Channel
0–7	8	9–16	17–23

Bits	Significance
0-7	The received character (when responding to a receive interrupt signal) or zeros (when responding to any other interrupt signal).
8(DO)	This bit is a 1 if a data overrun (rate error) has occurred; otherwise, this bit is a 0. A data overrun occurs when the computer fails to remove the received character from the receive buffer (with a PIN instruction) before the following character begins to arrive.
9-16	Bits 9 through 16 are always zeros.
17-23	The address of the channel causing the interrupt condition.

This instruction must be executed as part of the response to a CTE interrupt condition; otherwise, an erroneous operation may occur.

TEST CTE CHANNEL

2	40	7	1	1	BE	CP	DR	Channel
0–2	3–8	9–11	12	13	14	15	16	17–23

This configuration of the SKS instruction is used to test the transmit buffer and the data set. The test and channel are determined by bits 14 through 23 of the SKS instruction as shown on the following page.

Bit(s)	Function (if the corresponding bit is a 1)
14(BE)	Test for transmit buffer empty
15(CP)	Test for carrier present in the data set
16(DR)	Test for data set ready
17-23	The address of the channel to be tested

Bits 14-16 can be combined to test more than one function (with a single instruction). If all the test functions selected by bits 14-16 are satisfied, the computer skips the next instruction in sequence and executes the following instruction; otherwise, the computer executes the next instruction in sequence.

CTE OPERATIONS

When the START button on the CPU control panel is pressed, all data sets are deactivated. The START button must be held depressed for several seconds to allow the data sets to release the telephone lines and for the carrier to disappear; otherwise, carrier-present interrupts may occur. After the program has activated the desired channels, the corresponding data sets will go to the active mode to await an incoming call. Upon receipt of a call, the carrier-on interrupt signal for that channel will be issued. Thereafter, the interface controller will issue a receive interrupt signal each time the scanner encounters a raised receive flag (indicating that a character is in the receive buffer).

If the receive character is an end-of-message character, a different interrupt can be issued. This feature is a hardware option. If this feature is not present, the end-of-message character is treated the same as any other character.

After transmitting characters, the interface controller will issue a transmit interrupt signal each time a raised transmit flag is encountered, assuming that the transmit buffer is clear and the interrupt specification (POT instruction) requested an interrupt at the completion of transmission. The interface controller will also issue a carrier-on or carrier-off interrupt signal whenever the scanner encounters a data set whose carrier has changed from one state (on or off) to the other.

CTE INTERRUPTS

When responding to any CTE interrupt condition, the interrupt-servicing routine must perform an EOM-PIN sequence to determine the address of the channel causing the interrupt. When transmitting a character at an arbitrary time (i.e., not in response to an interrupt), the EOM-POT sequence should be preceded by a transmit buffer empty test to assure that the transmit buffer is not busy. Alternatively, the transmit interrupt for that channel could be enabled (with bit 8 of the word transmitted by the POT instruction equal to a 1). In the latter instance, the transmit interrupt signal will occur if (or when) the transmit buffer is empty. If a character is transmitted to a particular channel before it becomes ready, the preceding character (still being shifted out) will be destroyed.

When a particular channel is deactivated, it should not be reactivated until after the carrier-off interrupt signal for that channel is issued. At this time, the data set has released the telephone line and the channel can be reactivated, if desired, to await another call. Upon receipt of a call, the carrier-on interrupt signal will be issued. If the remote station should hang up, the carrier-off interrupt will be issued.

The scanner is halted every time that a raised receive flag, transmit-buffer-empty flag, carrier-on flag, or carrier-off flag is encountered. The scanner remains halted until the flag is cleared by the execution of the EOM-PIN sequence. During the time that the scanner is stopped, an interrupt signal corresponding to the type of flag encountered will be sent to the CPU. If more than one raised flag is encountered by the scanner, only the one of highest priority will result in an interrupt. The others will be ignored until the scanner has completed scanning all other channels. The receive flag will be given highest priority, followed by the transmit flag, the carrier-on flag, and the carrier-off flag.

For worst-case timing considerations, only one flag per channel is processed per scan cycle, and the longest interrupt-servicing routine determines the worst-case scan cycle time. A data overrun may occur if the scan cycle time exceeds approximately 27 milliseconds (15,400 computer cycles).

APPENDIX A CONVERSION TABLES

XDS 940 INTERNAL, ASCII, TELETYPE, LINE PRINTER, AND CARD CODES

INT	ASCII	TTY	LP	CARDS	INT	ASCII	TTY	LP	CARDS
00	40				40	100	@	✓	78
01	41	!	!	-0	41	101	A	A	+1
02	42	"	'	84	42	102	B	B	+2
03	43	#	‡	+78	43	103	C	C	+3
04	44	$	$	-38	44	104	D	D	+4
05	45	%	~	085	45	105	E	E	+5
06	46	&	△	-78	46	106	F	F	+6
07	47	'	'	84	47	107	G	G	+7
10	50	(	(	048	50	110	H	H	+8
11	51	)	)	+48	51	111	I	I	+9
12	52	*	*	-48	52	112	J	J	-1
13	53	+	+	+	53	113	K	K	-2
14	54	,	,	038	54	114	L	L	-3
15	55	-	-	-	55	115	M	M	-4
16	56	.	.	+38	56	116	N	N	-5
17	57	/	/	01	57	117	O	O	-6
20	60	0	0	0	60	120	P	P	-7
21	61	1	1	1	61	121	Q	Q	-8
22	62	2	2	2	62	122	R	R	-9
23	63	3	3	3	63	123	S	S	02
24	64	4	4	4	64	124	T	T	03
25	65	5	5	5	65	125	U	U	04
26	66	6	6	6	66	126	V	V	05
27	67	7	7	7	67	127	W	W	06
30	70	8	8	8	70	130	X	X	07
31	71	9	9	9	71	131	Y	Y	08
32	72	:	:	58	72	132	Z	Z	09
33	73	;	;	-68	73	133	[	[	+58
34	74	<	<	+68	74	134	\	\	068
35	75	=	=	38	75	135	]	]	-58
36	76	>	>	68	76	136	↑	‡	082
37	77	?	?	+0	77	137	←	⧻	087

SPECIAL CODES

INTERNAL	ASCII	CONTROL	FUNCTION
141	1	A	SOM
142	2	B	EOA
143	3	C	EOM
144	4	D	EOT
145	5	E	WRU
146	6	F	RU
147	7	G	BELL
151	11	I	TAB
152	12	J	LINE FEED
153	13	K	VT
154	14	L	FORM
155	15	M	RETURN
161	21	Q	X - ON
162	22	R	TAPE
163	23	S	X - OFF
164	24	T	TAPE
165	25	U	ESCAPE
166	26	V	SPACE
167	27	W	
170	30	X	
171	31	Y	
172	32	Z	

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

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

	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

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

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

	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

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	3072
to	to
6777	3583
(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
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	4022	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	3584
to	to
7777	4095
(Octal)	(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	.426781
.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

APPENDIX B. TWO'S COMPLEMENT ARITHMETIC

XDS computer systems hold negative numbers in memory in binary two's complement form. The two's complement of a binary number is formed by adding one to the one's complement (logical inverse) of the number. This convention allows the sign of a number to be used as an integral part of the number in all arithmetic operations and obviates the need for keeping track of a detached sign with computer logic.

In XDS systems, the sign bit is in the first bit position to the left of the most significant magnitude bit. Thus, if an XDS computer word was only 6 bits long instead of 24, some common decimal values would be represented in binary format as follows:

Decimal Number	Octal Equivalent	Complement Plus 1	Binary Equivalent
3	03	-	000 011
2	02	-	000 010
1	01	-	000 001
0	00	-	000 000
-1	(-)01	77	111 111
-2	(-)02	76	111 110
-3	(-)03	75	111 101
31	37	-	011 111
-31	(-)37	41	100 001

This table suggests the following algorithms:

1. To find the binary, two's complement of a negative decimal number:

   a. Find the octal equivalent of the absolute of the number.

   b. Form the complement and add one.

   c. Express as a binary number.

The result is the binary, two's complement equivalent.

2. To find the decimal equivalent of a binary two's complement number:

   a. Express as an octal number.

   b. Subtract one and form the complement.

   c. Find the decimal equivalent.

   The negative of the result is the decimal equivalent.

The following examples show how two's complement numbers automatically yield the correct result when used arithmetically in the computer:

Decimal Number	Binary Equivalent
+20	010 100
-03	111 101
+17	1 010 001 = $21_8 = 17_{10}$
	└—lost carry

Note that the carry out of the most significant (sign bit) position is lost. Nevertheless, the value remaining is the correct answer.

Decimal Number	Binary Equivalent
-32	100 000
+24	011 000
- 8	111 000 = $(-)10_8 = -8_{10}$

When performing additions or subtractions in the computer, carries out of the sign bit do not always signify a true overflow condition or cause the OVERFLOW indicator to be set. In an addition, it is impossible to produce an overflow if the signs of the operands are unlike. The computer sets the OVERFLOW indicator in an addition only when the signs of the two operands are the same but the sign of the result is opposite. In a subtraction, which in the computer is accomplished by forming the two's complement of the subtrahend and then adding to the minuend, the test for overflow is similar to that for addition. That is, overflow occurs when both numbers have the same sign after the subtrahend has been complemented but the sign of the result is opposite.

APPENDIX C INSTRUCTION LIST

Mnemonic	Code	Name	Page
ABC	0 46 00005	Copy A into B, Clear A	24
ADC	57	Add with Carry	20
ADD	55	Add	20
ADM	63	Add to Memory	20
AIR	0 02 20020	Arm Interrupts	13
ALCW	0 02 50000	Alert Channel W	37
ASCW	0 02 12000	Alert to Store Address in Channel W	38
BAC	0 46 00012	Copy B into A, Clear B	24
BPT1	0 40 20400	Test Breakpoint 1	30
BPT2	0 40 20200	Test Breakpoint 2	30
BPT3	0 40 20100	Test Breakpoint 3	30
BPT4	0 40 20040	Test Breakpoint 4	30
BRI	11	Branch and Return from Interrupt Routine	26
BRM	43	Mark Place and Branch	25
BRR	51	Return Branch	26
BRU	01	Branch Unconditionally	25
BRX	41	Increment Index and Branch	25
CAB	0 46 00004	Copy A into B	23
CATW	0 40 14000	Channel W Active Test	44
CAX	0 46 00400	Copy A into Index	24
CBA	0 46 00010	Copy B into A	24
CBX	0 46 00020	Copy B into Index	24
CETW	0 40 11000	Channel W Error Test	44
CITW	0 40 10400	Channel W Interrecord Test	44
CLA	0 46 00001	Clear A	23
CLB	0 46 00002	Clear B	23
CLAB	0 46 00003	Clear AB	23
CLEAR	2 46 00003	Clear A, B, and Index	23
CLX	2 46 00000	Clear Index	23
CNA	0 46 01000	Copy Negative into A	25
CXA	0 46 00200	Copy Index into A	24
CXB	0 46 00040	Copy Index into B	24
CZTW	0 40 12000	Channel W Zero Count Test	44
DIR	0 02 20004	Disable Interrupt System	13
DIV	65	Divide	22
DISW	0 02 00000	Disconnect Channel W	38
EAX	77	Copy Effective Address into Index	20
EIR	0 02 20002	Enable Interrupt System	13
EOD	06	Energize Output D	36
EOM	02	Energize Output M	36
EOR	17	Exclusive OR	23
ETR	14	Extract	22
EXU	23	Execute	30
HLT	00	Halt	30
IDT	0 40 20002	Interrupt Disabled Test	13
IET	0 40 20004	Interrupt Enabled Test	13
LCY	0 67 20xxx	Left Cycle AB	29
LDA	76	Load A	19
LDB	75	Load B	19
LDE	0 46 00140	Load Exponent	25
LDX	71	Load Index	19
LRSH	0 66 24xxx	Logical Right Shift AB	29
LSH	0 67 00xxx	Left Shift AB	29
MIN	61	Memory Increment	21
MIW	12	Memory into W Buffer	47
MRG	16	Merge	22
MUL	64	Multiply	21

Mnemonic	Code	Name	Page
NOD	0 67 10xxx	Normalize and Decrement Index	29
NOP	20	No Operation	30
OTO	0 22 00100	Overflow Test Only	31
OVT	0 22 00101	Overflow Test and Reset	31
PIN	33	Parallel Input	37
POT	31	Parallel Output	37
RCY	0 66 20xxx	Right Cycle AB	29
REO	0 22 00010	Record Exponent Overflow	31
ROV	0 22 00001	Reset Overflow Indicator	31
RSH	0 66 00xxx	Right Shift AB	29
SKA	72	Skip if Memory and A do not Compare Ones	27
SKB	52	Skip if Memory and B do not Compare Ones	28
SKD	74	Difference Exponents and Skip	28
SKE	50	Skip if A Equals Memory	27
SKG	73	Skip if A Greater than Memory	27
SKM	70	Skip if A Equals Memory on B Mask	27
SKN	53	Skip if Memory Negative	28
SKR	60	Reduce Memory, Skip if Negative	28
SKS	40	Skip if Signal not Set	36
STA	35	Store A	19
STB	36	Store B	19
STE	0 46 00122	Store Exponent	24
STX	37	Store Index	19
SUB	54	Subtract	21
SUC	56	Subtract with Carry	21
TOPW	0 02 14000	Terminate Output on Channel W	38
WIM	32	W Buffer into Memory	47
XAB	0 46 00014	Exchange A and B	24
XEE	0 46 00160	Exchange Exponents	25
XMA	62	Exchange Memory and A	20
XXA	0 46 00600	Exchange Index and A	24
XXB	0 46 00060	Exchange Index and B	24

APPENDIX D 930 COMPATIBILITY

The 'XDS 940 central processor is an extension of the XDS 930. This appendix describes the difference between the operation of the 930 and the operation of the 940 in each of its three modes.

NORMAL MODE

When the 940 is started in the normal manner by pushing the START button on the control panel, it automatically enters the normal mode. In this mode, the 940 is almost completely compatible with all 930 software. There are four differences between the 940, operating in the normal mode, and the 930; however, none of these differences affect the operation of the standard 930 software. These differences are:

1. A new instruction, designated by the mnemonic BRI and utilizing operation code 11 (which is an undefined operation code in the 930), has been added and is operative in the normal mode and the monitor mode. The description of this instruction appears in Section 3, "Branch Instructions".

2. A new instruction utilizing operation code 22 (which is an undefined operation code in the 930) has also been added and is operative in all 940 modes. This instruction performs four functions, as given by the following mnemonics and internal octal configurations:

Mnemonic	940 Configuration	930 Configuration
OVT	0 22 00101	0 40 20001
REO	0 22 00010	0 02 20010
ROV	0 22 00001	0 02 20001
OTO	0 22 00100	none

   The first three functions are identical to the standard 930 instructions with the same mnemonics. The 930 configurations of these instructions are operative; thus, there are two different methods for invoking these functions with the 940 while in the normal or monitor modes. The OTO instruction is described in Section 3, "Overflow Instructions".

3. A new configuration of the EOM instruction has been added to the 940 so that it can effect a program-controlled transition from the normal mode to the monitor mode. The octal configuration of this EOM is 0 02 22000 (see Section 2, "Mode-changing Capability.

4. The interlace word for the Data Multiplexing System is structured differently from that of a standard 930 (the count field only contains eight bits, permitting a maximum record size of 256 words). This is necessary so that the data address field can contain 16 bits and permit addressing of the full 64K-word 940 memory. (See Section 4, "Data Multiplexing System".)

MONITOR AND USER MODES

In addition to the differences between the 930 and 940 operating in the normal mode, several operations are effective in both monitor and user modes that are not effective in the normal mode; they include the following:

1. An EXECUTE instruction (or a long chain of EXECUTEs) is aborted in response to an interrupt request. This assures that the acknowledgement of an interrupt request is not excessively delayed because of a long chain or possibly an infinite loop of EXECUTE instructions. This process is effected by terminating the EXECUTE instruction and acknowledging the highest priority interrupt request. The P register contains the address of the terminated instruction; thus, the normal interrupt routine exit will return to the interrupted instruction, which will be restarted. Similarly, the execution of an instruction involving indirect addressing is interrupted when an interrupt request occurs during the indirect addressing phase of the execution. An interrupt request is also acknowledged at the completion of a BRX instruction that calls for a branch. In this case, the P counter contains the location specified by the branch.

2. A memory mapping technique that provides for dynamic relocation of programs, for fragmentation of memory, and for two modes of memory protection is included in the 940 (see Section 2, "Memory Access Control").

3. The interrupt-clearing function of the BRU instruction (when coded for indirect addressing) is inhibited.

MONITOR MODE

In addition to the above listed differences between the 930 and the 940 operating in the user and monitor modes, there are several other differences unique to the monitor mode. These additional differences are:

1. The execution of an instruction in which the content of bit position 0 (the sign bit) is a 1 causes address mapping through the user map to apply for that instruction. Mapping through the user map will also apply to any instruction for which the sign bit is a 1 in any word fetched during the determination of an effective address. More precisely, mapping through the user map becomes effective when a sign bit of 1 is detected, and the computer will remain in this mode for the duration of the current instruction. Thus, if the sign bit is a 1 in a word fetched during indirect addressing, all further memory references made by this instruction will be mapped through the user memory map.

2. The 940 contains a partially implemented monitor map. The monitor map is described in Section 2, "Memory Access Control".

3. The return address associated with the BRM, BRR, and POP instructions is different in the monitor mode

to allow return to the calling or interrupted program irrespective of the mode in which the computer was operating at the time of the call or interruption. The differences are:

a. The overflow indicator is stored in bit position 2 rather than in the sign bit position.

b. The sign bit is set to 1 if the return address is to be mapped through the user map; otherwise, the sign bit is set to 0. An example of the former is the occurrence of an interrupt or SYSPOP when the computer is in the user mode.

4. In addition to performing all its normal-mode operations, the execution of an EAX instruction in the monitor mode will cause bit position 0 of the index register to be set to 1 if the effective address is mapped through the user map. If the effective address is not mapped through the user map, bit position 0 of the index register is reset to 0.

5. A special monitor-to-user transition trap function has been incorporated. This trap may be enabled by a program operating in the monitor mode with a specific configuration of the EOM instruction described below, and is automatically disabled whenever it is invoked. It is also disabled whenever any other trap is invoked. If this trap is enabled, the 940 will trap to location 44B whenever a switch from the monitor mode to the user mode is attempted (see Section 2, "Trap System").

6. Special configurations of the EOM instruction have been incorporated so that the monitor-mode program may be entered and so that it may control its environment. These EOMs have the following octal configurations and perform the following functions:

Configuration	Function
0 02 20400	Clear and select RL1 for loading
0 02 21000	Clear and select RL2 for loading
0 02 21400	Clear and select RL4 for loading
0 02 22000	Perform the transition from normal to monitor mode
0 02 22400	Enable the monitor-to-user transition trap.

All these configurations of the EOM instruction may also be executed in the normal mode. In the first three cases, the EOM must be followed by a POT instruction that loads the memory map register (RL1, RL2, or RL4) with the appropriate word from memory (see Section 2, "Memory Address Control").

USER MODE

In addition to the above listed differences between the 930 and the 940 operating in the user and monitor modes, there are three characteristics unique to the user mode. These characteristics are:

1. In the user mode, memory mapping through the user memory map is always effective (see Section 2, "Memory Access Control").

2. A set of instructions (referred to as "privileged" instructions) is prohibited in the user mode. The set of privileged instructions includes the following:

   a. all input/output instructions (including EOM, EOD, and SKS)

   b. HALT (HLT) and BRANCH AND RETURN FROM INTERRUPT ROUTINE (BRI)

   c. all undefined operation codes

   Any attempt to execute a privileged instruction while the computer is in the user mode results in a trap to location 40_8 (see Section 2, "Trap System").

3. In the user mode, the execution of a Programmed Operator (POP) instruction with bit 0=0 and bit 2=1 causes a transfer to the user-mapped location designated by bits 2 through 8 with the link word being stored in user-mapped location 0. Insofar as the program is concerned, this operation is identical to that performed by the 930. However, the execution of a POP instruction with bit 0=1 and bit 2=1 causes a transfer to the absolute (nonmapped) location designated by bits 2 through 8 with the link word being stored in absolute location 0. In addition, this latter case causes a transition from the user mode to the monitor mode and causes the sign bit position of absolute location 0 to be set to 1, indicating that the return address is to be user mapped. This form of POP instruction is designated as SYSPOP, since it makes the system programmed operators directly available to the user-mode program without requiring the monitor-mode program to intervene.

SUMMARY

The XDS 940 is basically an augmented XDS 930 with very few changes to the 930 subset. With the exception of the improved interrupt routine exit and the alternate overflow indicator instructions, all instruction differences are associated with the new features; namely, the computer modes and the memory address mechanism. However, numerous logic changes to the 930 subset exist, although they are not discernible by the programmer.

INDEX

XDS 940 OPERATION CODES

Code	Mnemonic	Name	Page
00	HLT	Halt	30
01	BRU	Branch Unconditionally	25
02	EOM	Energize Output M	36
0 02 00000	DISW	Disconnect Channel W	38
0 02 12000	ASCW	Alert to Store Address in Channel W	38
0 02 14000	TOPW	Terminate Output on Channel W	38
0 02 20001		Reset Overflow Indicator	31
0 02 20002	EIR	Enable Interrupt System	13
0 02 20004	DIR	Disable Interrupt System	13
0 02 20010		Record Exponent Overflow	31
0 02 20020	AIR	Arm Interrupts	13
0 02 20400		Clear and Select RL1 for Loading	9
0 02 21000		Clear and Select RL2 for Loading	9
0 02 21400		Clear and Select RL4 for Loading	9
0 02 22000		Perform Transition to Monitor Mode	10
0 02 22400		Enable the Monitor-to-User-Transition Trap	17
0 02 50000	ALCW	Alert Channel W	37
06	EOD	Energize Output to Direct Access Channel	36
0 06 200SR		Set Extension Register	31
10	MIY	Memory into Y Buffer	50
11	BRI	Branch and Return from Interrupt Routine	26
12	MIW	Memory into W Buffer	50
13	POT	Parallel Output	37
14	ETR	Extract	22
16	MRG	Merge	22
17	EOR	Exclusive OR	23
20	NOP	No Operation	30
0 22 00001	ROV	Reset Overflow Indicator	31
0 22 00010	REO	Record Exponent Overflow	31
0 22 00100	OTO	Overflow Indicator Test Only	31
0 22 00101	OVT	Overflow Indicator Test and Reset	31
23	EXU	Execute	30
30	YIM	Y Buffer into Memory	50
32	WIM	W Buffer into Memory	50
33	PIN	Parallel Input	37
35	STA	Store A	19
36	STB	Store B	19
37	STX	Store Index	19
40	SKS	Skip if Signal not Set	36
0 40 10400	CITW	Channel W Interrecord Test	44
0 40 11000	CETW	Channel W Error Test	44
0 40 12000	CZTW	Channel W Zero Count Test	44
0 40 14000	CATW	Channel W Active Test	44
0 40 20001		Overflow Indicator Test and Reset	31
0 40 20002	IDT	Interrupt Disabled Test	13
0 40 20004	IET	Interrupt Enabled Test	13
0 40 20040	BPT4	Test Breakpoint 4	30
0 40 20100	BPT3	Test Breakpoint 3	30
0 40 20200	BPT2	Test Breakpoint 2	30
0 40 20400	BPT1	Test Breakpoint 1	30
0 40 4000T		Extension Register Test	31
41	BRX	Increment Index and Branch	25
43	BRM	Mark Place and Branch	25
0 46 00001	CLA	Clear A	23
0 46 00002	CLB	Clear B	23
0 46 00003	CLR	Clear AB	23
0 46 00004	CAB	Copy A into B	23
0 46 00005	ABC	Copy A into B, Clear A	24
0 46 00010	CBA	Copy B into A	24
0 46 00012	BAC	Copy B into A, Clear B	24
0 46 00014	XAB	Exchange A and B	24
0 46 00020	CBX	Copy B into Index	24
0 46 00040	CXB	Copy Index into B	24
0 46 00060	XXB	Exchange Index and B	24
0 46 00122	STE	Store Exponent	24
0 46 00140	LDE	Load Exponent	25
0 46 00160	XEE	Exchange Exponents	25
0 46 00200	CXA	Copy Index into A	24
0 46 00400	CAX	Copy A into Index	24
0 46 01000	CNA	Copy Negative into A	25
2 46 00000	CLX	Clear Index	23
2 46 00003	CLEAR	Clear A, B, and Index	23
50	SKE	Skip if A Equals Memory	27
51	BRR	Return Branch	26
52	SKB	Skip if B and Memory do not Compare Ones	28
53	SKN	Skip if Memory Negative	28
54	SUB	Subtract	21
55	ADD	Add	20
56	SUC	Subtract with Carry	21
57	ADC	Add with Carry	20
60	SKR	Reduce Memory, Skip if Negative	28
61	MIN	Memory Increment	21
62	XMA	Exchange Memory and A	20
63	ADM	Add to Memory	20
64	MUL	Multiply	21
65	DIV	Divide	22
0 66 00xxx	RSH	Right Shift AB	29
0 66 20xxx	RCY	Right Cycle AB	29
0 66 24xxx	LRSH	Logical Right Shift AB	29
0 67 00xxx	LSH	Left Shift AB	29
0 67 10xxx	NOD	Normalize and Decrement Index	29
0 67 20xxx	LCY	Left Cycle AB	29
70	SKM	Skip if A Equals Memory on B Mask	27
71	LDX	Load Index	19
72	SKA	Skip if A and Memory do not Compare Ones	27
73	SKG	Skip if A Greater than Memory	27
74	SKD	Difference Exponents and Skip	28
75	LDB	Load B	19
76	LDA	Load A	19
77	EAX	Copy Effective Address into Index	20

XDS

Xerox Data Systems 701 South Aviation Blvd. / El Segundo, California 90245 (213) 772-4511 / Cable SCIDATA / Telex 674839 / TWX 910-325-6908

EASTERN TECHNOLOGY CENTER
12150 Parklawn Drive
Rockville, Maryland 20852
(301) 933-5900

PRINTED CIRCUITS DEPT.
600 East Bonita Avenue
Pomona, Calif. 91767
(714) 624-8011

TECHNICAL TRAINING
5250 West Century Blvd.
Los Angeles, Calif. 90045
(213) 772-4511

INTERNATIONAL MANUFACTURING SUBSIDIARY

Scientific Data Systems Israel, Ltd.
P.O. Box 5101
Haifa, Israel
04-530253, 04-64589
Telex 922 4474

SALES OFFICES

Western Region

Building Arts Bldg.
Suite G100
5045 N. 12th St.
Phoenix, Arizona 85014
(602) 264-9324

1360 So. Anaheim Blvd.
Anaheim, Calif. 92805
(714) 774-0461

*5250 West Century Blvd.
Los Angeles, Calif. 90045
(213) 772-4511

Vista Del Lago Office Center
122 Saratoga Avenue
Santa Clara, Calif. 95050
(408) 246-8330

Wells Fargo Building
Suite 500
21535 Hawthorne Blvd.
Torrance, Calif. 90503
(213) 542-5561

3333 South Bannock
Suite 400
Englewood, Colo. 80110
(303) 761-2645

320 Ward Avenue
Honolulu, Hawaii 96814
(808) 531-8257

Fountain Professional Bldg.
9004 Menaul Blvd., N.E.
Albuquerque, N.M. 87112
(505) 298-7683

El Paso Natural Gas Bldg.
Suite 201
315 E. 2nd South Street
Salt Lake City, Utah 84111
(801) 322-0501

400 Building
Suite 415
400 108th Avenue N.E.
Bellevue, Wash. 98004
(206) 454-3991

Midwestern Region

*2720 Des Plaines Avenue
Des Plaines, Illinois 60018
(312) 824-8147

Clausen Bldg., Suite 310
16000 W. Nine Mile Road
Southfield, Michigan 48075
(313) 353-7360

4410 Woodson Road
Suite 111
St. Louis, Missouri 63134
(314) 423-6200

One Erieview Plaza
13th Floor
Cleveland, Ohio 44114
(216) 522-1850

Seven Parkway Center
Suite 238
Pittsburgh, Pa. 15220
(412) 921-3640

Southern Region

State National Bank Bldg.
Suite 620
200 W. Court Square
Huntsville, Alabama 35801
(205) 539-5131

Orlando Executive Center
1080 Woodcock Road
Orlando, Florida 32803
(305) 841-6371

2964 Peachtree Road, N.W.
Suite 350
Atlanta, Georgia 30305
(404) 261-5323

Jefferson Bank Bldg.
Suite 720
3525 N. Causeway Blvd.
Metairie, Louisiana 70002
(504) 837-1515

4920 S. Lewis Avenue
Suite 103
Tulsa, Oklahoma 74105
(918) 743-7753

8383 Stemmons Freeway
Suite 233
Dallas, Texas 75247
(214) 637-4340

*2300 West Loop South
Suite 150
Houston, Texas 77027
(713) 623-0510

Eastern Region

10227 Wincopin Circle
Suite 716
Columbia, Maryland 21043
(301) 730-4900

20 Walnut Street
Wellesley Hills, Mass. 02181
(617) 237-2300

Brearley Office Building
190 Moore Street
Hackensack, N. J. 07601
(201) 489-0100

The Fortune Building
280 North Central Avenue
Hartsdale, New York 10530
(914) 948-2929

*1301 Avenue of the Americas
New York City, N.Y. 10019
(212) 765-1230

673 Panorama Trail West
Rochester, New York 14625
(716) 586-1500

P.O. Box 168
535 Pennsylvania Ave.
Ft. Washington Industrial Park
Ft. Washington, Pa. 19034
(215) 643-4250

Kogerama Building
Suite 212
No. 1 Tidewater
Executive Center
Norfolk, Virginia 23502
(703) 497-6811

Washington (D.C.) Operations

*2351 Research Blvd.
Rockville, Maryland 20850
(301) 948-8190

Canada

864 Lady Ellen Place
Ottawa 3, Ontario
(613) 722-8387

1009 7th Avenue, S.W.
Calgary 2, Alberta
(403) 265-8134

280 Belfield Road
Rexdale 605, Ontario
(416) 677-8422

1901 North Service Road
Trans Canada Highway
Dorval, P.Q.
(514) 683-3755

INTERNATIONAL OFFICES & REPRESENTATIVES

European/African Headquarters

Scientific Data Systems
I.L.I. House, Olympic Way
Wembley Park (London)
Middlesex, England
(01) 903-2511, Telex 27992

Sweden

Nordisk Elektronik AB
Stureplan 3
Stockholm 7
(08) 24 83 40

Denmark

A/S Nordisk Elektronik
Danasvej 2
Copenhagen V
EVA 8285/EVA 8238

Norway

Nordisk Elektronik (Norge) A/S
Middelthunsgt. 27
Oslo 3
(2) 60 25 90

France

Compagnie Internationale pour l'Informatique, C.I.I.
66, Route de Versailles
78-Louveciennes
Yvelines
951 86 00 (Paris area)

Israel

Elbit Computers Ltd.
Subsidiary of Elron Electronic Industries Ltd.
88 Hagiborim Street
Haifa
6 4613

*Regional Headquarters

400 Building, Suite 415, 400 108th Avenue, N.E. / Bellevue, Washington 98004 / (206) 454-3991 / TWX: 910-443-2440

90 06 40C