GA34-0003-7
File No. S7-00

Systems

IBM System/7
Functional Characteristics

IBM

GA34-0003-7
File No. S7-00

Systems

IBM System/7
Functional Characteristics

IBM

Eighth Edition (February 1976)

This is a reprint of GA34-0003-6 incorporating changes released in Technical Newsletter GN34-0290, dated October 10, 1975.

Changes are periodically made to the information herein; before using this publication in connection with the operation of IBM systems, refer to the latest *System/7 Bibliography*, GC20-1737 for the editions that are applicable and current.

Text for this manual has been prepared with the IBM Selectric® Composer.

Some illustrations in this manual have a code number in the lower corner. This is a publishing control number and is not related to the subject matter.

Requests for copies of IBM publications should be made to your IBM representative or the IBM branch office serving your locality.

A form for readers' comments is provided at the back of this publication. If the form has been removed, send your comments to IBM Corporation, Systems Publications, Department 27T, P.O. Box 1328, Boca Raton, Florida 33432. Comments become the property of IBM.

This reference manual is a source of information primarily for programming the applications of System/7 in machine language. The manual describes the operation and use of the machine instructions and the functions that are used to program the processor and I/O devices. Specific topics covered are:

- Processor functional characteristics
- Priority interruptions
- Instruction set
- I/O commands and device status
- Operator console
- Interval timers
- Asynchronous Communications Control Attachment
- Binary Synchronous Communications Adapter
- 1130 host attachment
- Operator station
- Analog input and output
- Digital input and output
- 2790 Control
- Disk storage module
- 5024 I/O Attachment Enclosure

The reference appendixes include summaries of instructions, I/O commands, and status words; a character code chart; and examples of how to convert a voltage value to or from binary data.

The reader must have a background in sensor-based data processing concepts and be familiar with the hexadecimal numbering system as used in IBM systems. The manual *Number Systems,* GC20-1618, contain information on hexadecimal numbering.

Prerequisite reading for this publication is *IBM System/7 System Summary,* GA34-0002. Information on customer interfaces to the I/O modules (analog, digital, and 2790 Control) appears in *IBM System/7 Installation Manual–Physical Planning,* GA34-0004.

This manual has been organized so that each chapter forms a module within the entire manual. Each chapter has its own front cover, table of contents, and page and figure numbers. A master table of contents appears at the beginning of the manual, and a master index is included at the end of the manual.

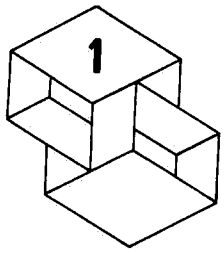
The modular presentation gives the user an option of using this manual in its published format, or of using the chapters in combination with information from other sources. That is, it may be desirable to insert one or more chapters from this manual into another publication.

The name of the manual appears at the foot of each left-hand page; the title of the chapter appears at the foot of each right-hand page. The user who separates the chapters should find this arrangement helpful for retrieving and reorganizing them.

Examples of data and instruction formats used throughout the manual give the binary bit settings and their equivalent hexadecimal values. Bit positions that can be set by the program or programmer are denoted by an X. Bit positions that are not used by a particular data or instruction format are denoted by *. The latter bit positions, however, must be set to 0 values.

Use of the leading P in the instruction mnemonics is required in all System/7 programs to be assembled on an 1130 or 1800 system, or using the System/7 Stand-Alone Assembler. The P or an X is always required in the Execute I/O instruction (PIO or XIO). The P is optional (except for PIO) when assembling on a System/360, System/370, or the System/7 Macro Assembler.

CHAPTER CONTENTS



SYSTEM/7 CONFIGURATIONS

The System/7 is designed so that both its functional and physical structure are highly modular. This permits great flexibility in designing configurations for a broad spectrum of applications. Furthermore, the modularity allows easy expansion of a small initial installation to a more powerful configuration as future needs require.

System/7 can operate as a standalone computing system, or it can function as a satellite processor linked to a host processor which is on-site or at a remote distance.

The System/7 is structured independently of any host processor, and is designed to provide modular data acquisition and modular sensor-based I/O functions independent of the host processor to which it may be linked. Therefore, the host processor can be modified or replaced without affecting the System/7. This gives an installation more flexibility in its overall system planning and minimizes the disturbance to its machine interfaces.

The System/7 can be configured in any one of the following ways:

1. A standalone system with one I/O module of any type contained in the IBM 5026 enclosure. (See Figure 1-1.)
2. A standalone system with as many as 11 I/O modules in any combination and a separate IBM 5024 attachment enclosure with a line printer and/or a 2502 card reader.
3. One or more of the configurations described in items 1 and 2 operating as satellite processors linked to the IBM System/360 (model 25 and larger), the IBM System/370 or the IBM 1800 Data Acquisition and Control System. To accomplish this, each System/7 must have the Asynchronous Communications Control Attachment (ACCA).
4. One or more of the configurations described in items 1 and 2 operating as satellite processors linked to the IBM System/370, the IBM System/3, or another System/7. To accomplish this, each System/7 must have the Binary Synchronous Communications Adapter (BSCA).
5. Any one of the configurations described in items 1 and 2 attached directly to an IBM 1130 Computing System via the 1130 Storage Access Channel.

When operating as a satellite processor or directly attached to the 1130 Storage Access Channel, it is possible to initial program load (IPL) the System/7 from the host processor.

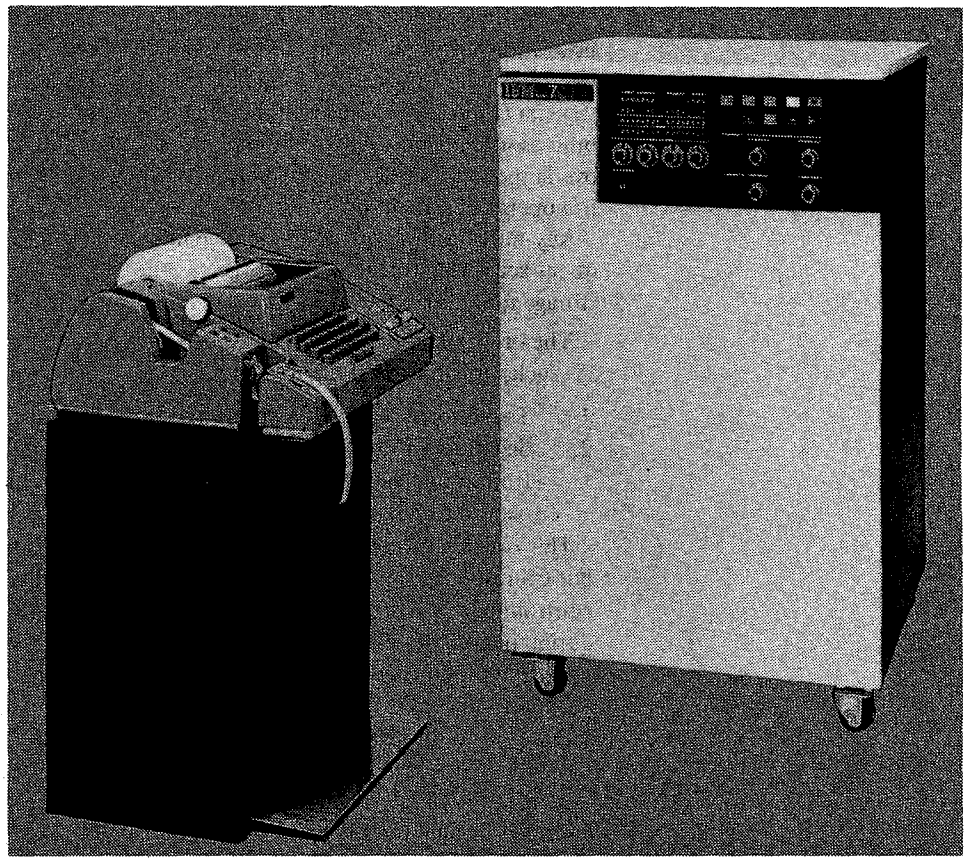
The IBM 5028 Operator Station is used for communication between the operator and the System/7. Input is presented via the keyboard or paper tape reader. Output is via the printer or paper tape punch. Preparation of programs and IPL of the system can be accomplished through the operator station.

IPL can also be accomplished through the IBM 5022 Disk Storage Module.

Processor Module

Every System/7 configuration has a 5010 Processor Module which controls the system, performing data processing operations and issuing I/O instructions to the I/O modules. The main elements of the processor module are a processor, a monolithic storage, and a channel.

The processor is a compact, 16-bit binary computer with a 400-nanosecond storage cycle time. Four priority interruption levels each with sixteen sublevels permit up to 64 independent interrupt servicing routines to be automatically accessed by the processor. Interruption priority level switching time is minimized by four sets of machine registers and program indicators, one set for each of the four interruption levels. Interruptions also occur for error conditions such as program check, machine check, and power/thermal failure conditions.



BR0712A

Figure 1-1. IBM 5028 Operator Station and IBM 5026 Enclosure Model A2

The monolithic storage is organized into words of 16 bits (two bytes), plus a parity bit for each byte. For the 5010 Processor Module models A and B, the smallest storage size is 2K (2,048) words. Additional storage can be obtained in increments of 2K words to a maximum of 16K (16,384) words. For the Model E, storage size begins at 16K (16,384) words and, in increments of 4K, ends at 64K (65,536) words. Power to the system must be maintained to keep data in storage.

The processor module contains two interval timers that function as separate program-controlled counters. Each is a 16-bit binary counter that decrements at 50-microsecond intervals. The processor module also contains the adapter for attaching the 5028 Operator Station. The model A or E processor module may also contain the Asynchronous Communications Control Attachment (ACCA) for communicating with a System/360, a System/370 or an 1800 Data Acquisition and Control System. The model A or E processor module may also contain the Binary Synchronous Communications Adapter (BSCA) for communicating with a System/3, a System/370, or another System/7. The ACCA and the BSCA are mutually exclusive within the model A or E processor module. The model B processor module contains an adapter for attaching the System/7 to the 1130 Storage Access Channel.

The channel is the communications link between the I/O devices and modules and the processor module. The channel interprets part of I/O instructions and provides some control over interruptions to the processor. During data transfer, each word (16 bits) of data must be transferred between the I/O device and the processor via a separate I/O instruction. This method of operation is called direct program control. No processing operations can take place while the data transfer is in progress.

When the optional cycle steal (CS) feature is installed in the 5010 Processor Module, however, the system can operate with a 5022 Disk Storage Module (also equipped with the cycle steal feature) in a mode known as cycle stealing. This is a method of data transfer (up to 3,072 words with a single I/O instruction) that significantly reduces processor load and response time required to handle data transfers. I/O operations are overlapped with processing operations so that processing operations can continue while I/O operations are in progress.

Any model of the processor module may accommodate the CS feature which allows direct access to storage, through the channel, by an I/O device (for example, the disk storage module).

The order of priorities in the CS feature is tailored to a process control environment. As such, priority in the channel is assigned as follows:

1. Top priority to the presentation of interruptions to the processor.
2. Secondary priority to execution of I/O commands, or polling for interruptions from modules and servicing CS requests. Executing commands and polling do not occur at the same time.

This order of priorities allows maximum system response to interruption conditions, but CS response capability is sacrificed to do this. Thus, if a nonbuffered subchannel is controlling a synchronous I/O module, (for example, a 5022 Disk Storage Module) during periods of high channel loading, a data service overrun may occur. This happens when the module's request for service exceeds the channel's capability to provide service.

In a typical operating environment, overrun conditions should be extremely rare. If they do occur, however, comprehensive status is provided to allow recovery.

Cycle steal operations between the processor module and the disk storage module are described in Chapter 15.

I/O Modules

I/O modules are physical structures housing the total logic and signal processing functions for I/O devices. Each I/O module connects to the processor through the system internal interface and channel. These I/O modules are:

- IBM 5014 Analog Input Module
- IBM 5012 Multifunction Module
- IBM 5013 Digital I/O Module
- IBM 5022 Disk Storage Module

Logic contained within an I/O module is divided into two parts:

1. Common controls.
2. Device logic and special controls that vary depending upon the type of I/O device.

The common control logic governs data transfers between the I/O modules and the processor module. Device logic and special controls govern data transfer between the device (for example, analog point, digital group, disk file) and the common control logic.

Sensor-Based Modules

The IBM 5014 Analog Input Module is available in 5 models (models B1, C1, D1, E1 and E2). Each model can have as many as 128 analog input points, but they operate at different scanning rates (number of points scanned per second). Each model is described more fully later in this manual.

The IBM 5012 Multifunction Module may contain the following features:

- As many as 32 analog input points. All points must be of the same speed. Performance of the analog input feature in this module is identical to that of the IBM 5014 Analog Input Module (models B1 and C1).

- One or two analog output points.
- As many as 128 digital input points, divided into eight 16-point groups. The first two groups can each have an optional process interrupt feature that generates an interruption from a change in status of a point. Each group provides 16 non-isolated contact sense points or 16 isolated points. The processor interrupt feature is available only with isolated points.
- As many as 64 digital output points, divided into four 16-point groups. Each group provides 16 isolated points or 16 non-isolated, solid-state switches for switching up to 450 ma. of user-supplied power.
- One 2790 Control feature for attaching a single loop of the IBM 2790 Data Communication System.

The IBM 5013 Digital I/O Module is available in one model (Model A1). Any combination of the following module features is available:

- Digital input
- Digital output
- 2790 Control
- Custom interface and custom attachment

The digital I/O and the 2790 Control features are identical to those in the 5012 Multifunction Module.

Each 5013 module can accommodate one custom interface. The number of custom attachments in the module, however, is limited by the physical size of the attachments.

For information on IBM data processing equipment that may be attached to the System/7, refer to *IBM System/7 129/5496/7431 Data Processing I/O Attachments General Information Manual RPQ Number D08147,* Order No. GA34-1514.

Disk Storage Module

The IBM 5022 Disk Storage Module is available in four models. Models 1 and 2 each contain two disks, one above the other. The upper disk is contained in a disk cartridge so that it can be used for offline storage. Disk storage module models 3 and 4 each contain only one disk, which cannot be used for offline storage.

All four models of the disk storage module read and write data on a magnetic surface at the same rate of speed. Models 1 and 3 differ from models 2 and 4, respectively, only in the time required to move the access mechanism. The disk storage module is described more fully in Chapter 15 of this manual.

Any model of the disk storage module can accommodate the optional disk cycle steal (CS) feature (the CS basic feature in the processor module is a prerequisite). The optional feature allows the transfer of up to 3,072 words to or from the disk with only one I/O command.

5024 I/O Attachment Enclosure

The IBM 5024 I/O Attachment Enclosure provides medium speed line printing and card read capabilities for the System/7 via a Line Printer and a 2502 Model A2 Card Reader. The 5024 differs from conventional System/7 I/O modules in that it resides outside of the 5026 within its own enclosure. The enclosure houses the power and logic necessary to control the printer and card reader mounted on it.

The 5024 can attach to the 5010E processor mounted in a 5026, A2, C3, or C6 enclosures. Any one of the three model configurations can be attached. The 5024 communicates with the System/7 through control logic in the 5010E processor. The System/7 communicates with the 5024 through set interrupt commands.The 5024 attachment is described in greater detail in Chapter 17 of this manual.

SYSTEM DATA FLOW

Data can enter or leave the System/7 through the operator station, the I/O modules, the 5024 or a host system.

Data entered through the operator station can consist of programs to be executed, data to be used by the program, or direct commands from the operator to the system. Data leaving the system through the operator station can consist of messages to the operator or data to be placed on the printer and/or paper tape.

Data values, control information, and status information (collectively called data) are transferred in both directions between the I/O modules (including the 5024) and the processor module. The System/7 processor can (1) evaluate and/or manipulate the data, (2) store the data in main storage, or (3) send the data to a host processor for processing and/or storage. Generally, a host system transmits to System/7 the control programs, data, and parameters within which certain processes are to be maintained. In turn, System/7 sends to the host system the collected data for processing, status information, or requests for new routines to handle situations which cannot be serviced by the routines that currently reside in the System/7.

POWER SYSTEM CONTROL

Power Control

Each 5026 Enclosure in a System/7 configuration contains its own power system. A 5026 Enclosure that contains a processor module is called a master unit; an enclosure that does not contain a processor module is called a remote unit. Power to remote units is normally controlled by the master unit. If, however, a remote unit loses power, its master unit does not lose power. A remote unit that loses power may give erroneous data, depending on the program that is executing at the time. The 5024 I/O Attachment Enclosure provides power for the line printer and 2502 card reader. The 5024's power is not controlled by the master unit.

Control Switches

The operator console on a master unit contains a power on/off switch and, optionally, an automatic restart switch. The power on/off switch controls the power state of all modules in the System/7 configuration with the execption of the 5024. The automatic restart switch controls the ability of the System/7 to restore power and perform an initial program load (IPL) when power is restored. The use of this switch is described later in this discussion under "Power Failure."

The operator console of a remote unit contains a power on/off switch and, optionally, an automatic restart switch. A local/remote switch is located behind the front cover panel immediately below the operator console. This switch is used primarily for servicing the enclosure and its I/O modules. When the switch is set to the remote position, power to the remote unit is controlled by the position of the power on/off switch on the master unit. When the switch is set to the local position, power to the remote unit is controlled by the position of its power on/off switch, regardless of the power state of the master unit. If power control of a remote unit is to be transferred from remote to local, the executing program should be modified prior to the actual transfer so that the modules in the remote unit are not addressed. This should avoid unnecessary errors being sent to the processor. Likewise, the program should be modified when power control is transferred from local to remote.

The power on/off switch on a remote unit can be used to remove power from the remote enclosure any time that the master unit is also powered on, regardless of the position of the local/remote switch in the remote unit.

Power Failure and Thermal Warning

Power Failure

The power failure detect feature in the System/7 presents a warning of possible power failure due to low voltage or loss of primary voltage in a 5026 Enclosure. Input power falling below 85 percent of nominal voltage actuates the power failure detection circuit. This circuit signals the processor that an imminent power shutdown in the enclosure is possible. The signal remains in effect for the duration of the low-voltage condition. If the master unit loses power, the entire system shuts down; if a remote unit loses power, only the remote unit shuts down.

With this feature, power can be automatically restored to the enclosure that sustained the power loss if the automatic restart switch on the operator console is turned on. When the switch is set in the on position and a power failure occurs, the modules in the enclosure are automatically reset when power is restored. If the failure occurs in a master unit, an automatic IPL takes place if the operator console and IPL device are properly set up. If the power failure and automatic restart occurs in a remote unit, no IPL takes place in the master unit.

Even though the power failure detect feature is installed on an enclosure, power is not automatically restored if the automatic restart switch is set in the off position. In that position, an operator must manually set the power switch to the off position and then back to the on position in order to restore power to the enclosure.

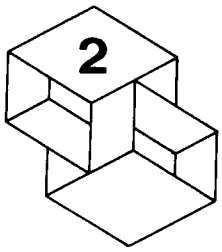
The power failure detection circuit signals the processor of an imminent power shutdown regardless of the setting of the automatic switch.

Thermal Warning

The System/7 operates in a temperature range from 40°F to 122°F (4.4°C to 50°C). For details on the temperature range for individual units, refer to the publication *IBM System/7 Installation Manual–Physical Planning*, Order No. GA34-0004. System/7 temperature sensing circuits are provided to remove power from the enclosure to prevent damage to the system in an over-temperature environment. Prior to shutting down the power system, the sensing circuits alert the processor that an imminent power shutdown is possible. The over-temperature condition is also indicated by the thermal light on the operator console.

Power can be restored when the thermal light goes off by turning the power switch to off and then to on. If the thermal condition that caused the power shutdown persists, a customer engineer must be called.

CHAPTER CONTENTS



PROCESSOR MODULE DATA FLOW

A general data flow for the processor module is shown in Figure 2-1. The 1130 attachment, if installed, can directly address System/7 storage.

Data flow within the processor module is shown in Figure 2-2. All System/7 commands are interpreted and executed by the processor. The first word of a command enters the operation (Op) register for decoding.

Storage is normally addressed from the processor by the storage address register (SAR). All data entering or leaving main storage must pass through the storage data register (SDR), with the exception of cycle steal.

The arithmetic and logic unit (ALU) performs addition; subtraction; the logical operations AND, OR, and exclusive OR; and address arithmetic. (Address arithmetic is explained in Chapter 4 under "Effective Address Generation.")

The Y register shifts data to the left or to the right, or acts as a buffer for one operand of the ALU.

Most registers in the processor are duplicated for each level to permit fast status switching. The registers associated with a particular level are used only when the processor is operating on that level.

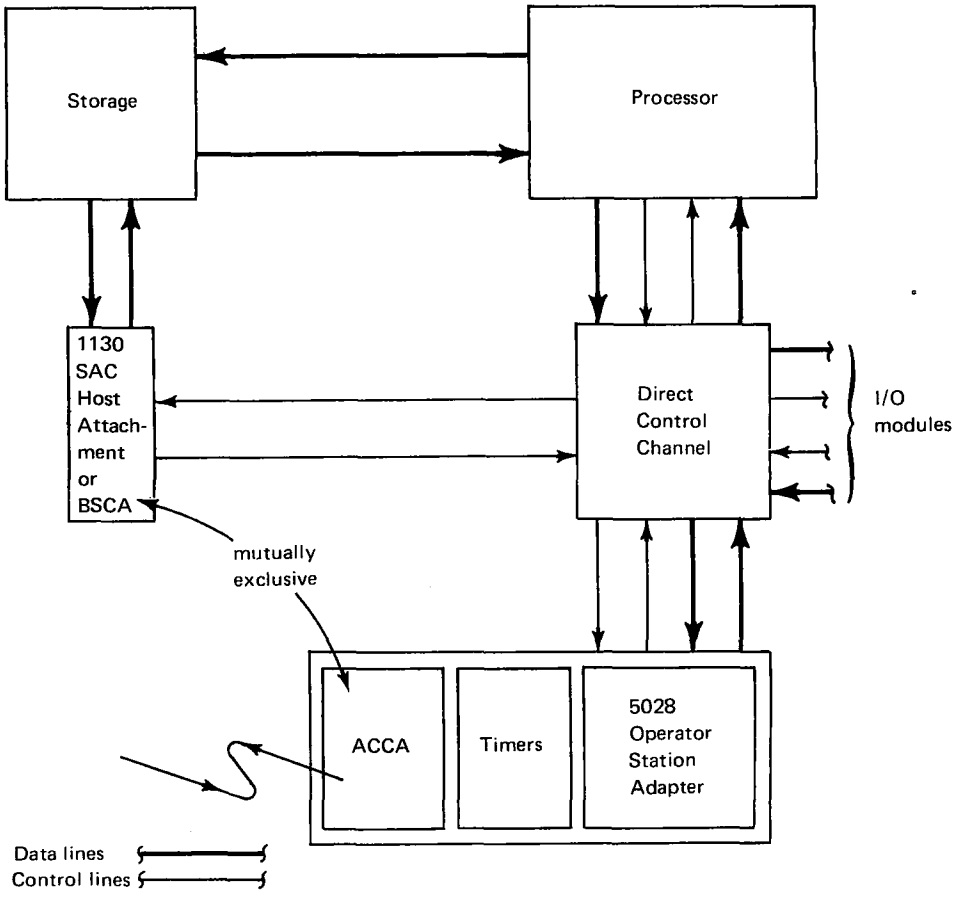


Figure 2-1 (Part 1 of 2). General data flow for 5010 Processor Module: without cycle steal (Part 1) and with cycle steal (Part 2)

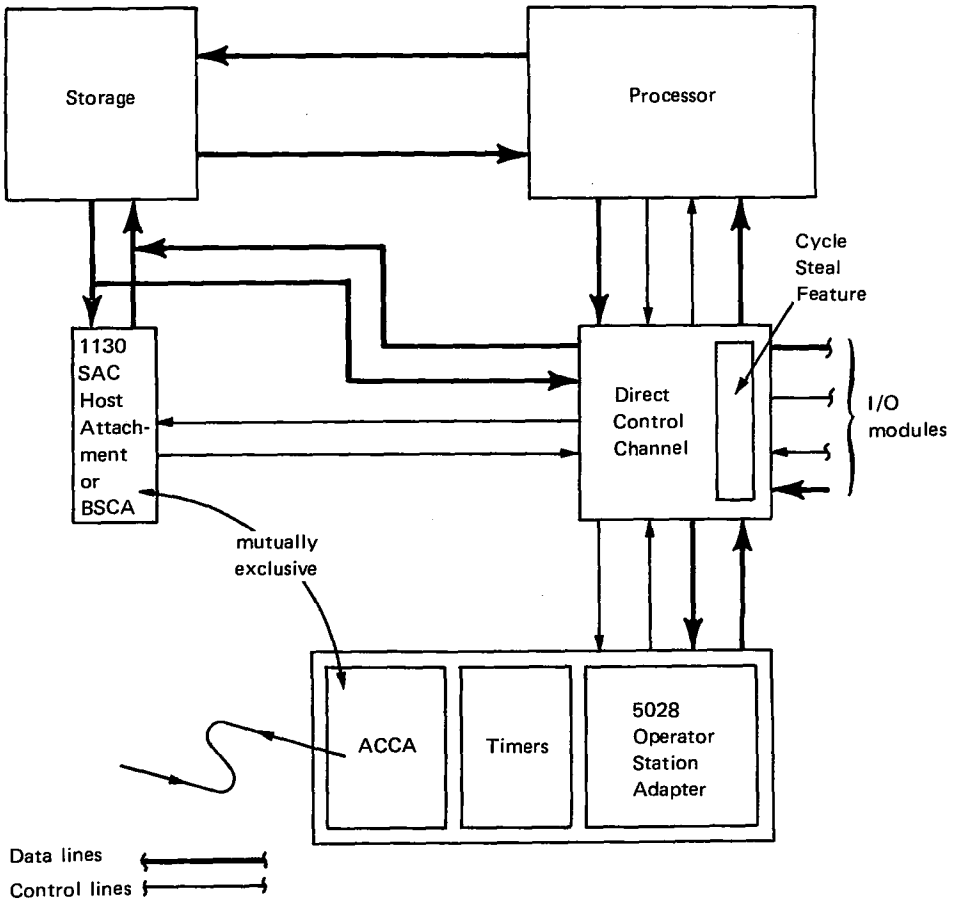
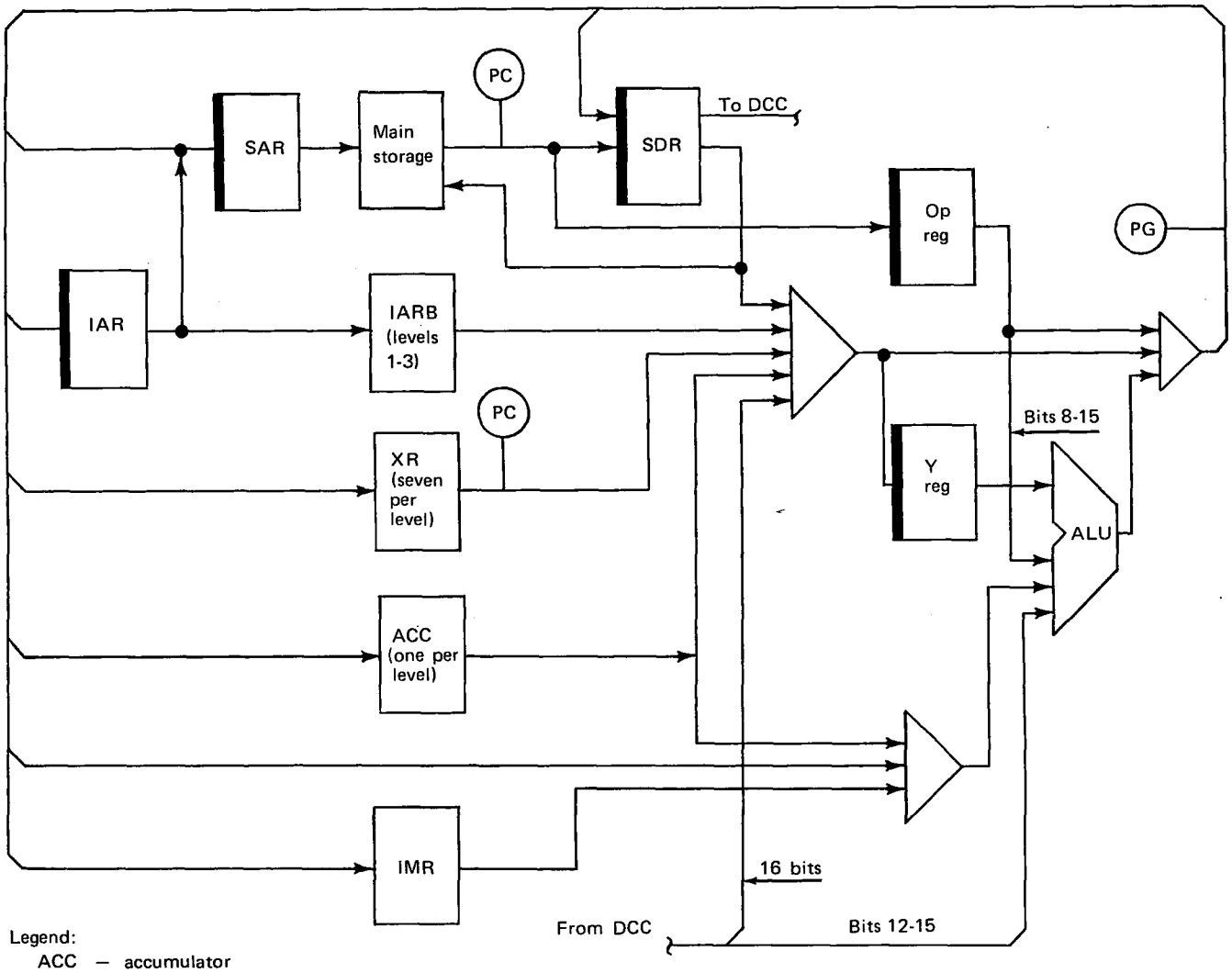


Figure 2-1 (Part 2 of 2). General data flow for 5010 Processor Module: without cycle steal (Part 1) and with cycle steal (Part 2)



Legend:

ACC – accumulator
ALU – arithmetic and logic unit
DCC – direct control channel
IAR – instruction address register
IARB – instruction address register (backup)
IMR – interruption mask register
PC – parity checked at this point
PG – parity generated at this point
SAR – storage address register
SDR – storage data register
XR – index register

Figure 2-2. Processor data flow

Processor States

Stop State

The processor enters the stop state and turns on the stop light when any one of the following occurs:

- The stop key on the System/7 console is depressed.
- The processor executes the stop (PSTP) instruction.
- The processor detects an error or a specified address, determined by switch settings on the operator console. (See "Check Control Switch" and "Rate Control Switch" in Chapter 5.)

The processor does not recognize interruption requests while it is in the stop state, but the requests are not lost. They are stacked and become pending, to be serviced when processing is resumed. The operator console functions can be used provided they are enabled by the console controls switch. When the start key is pressed, the processor exits from the stop state and returns to the state that was left when it entered the stop state. However, if the stop state is entered under control of the check control switch (stop on error), the system must be reset by the reset key in order to remove the error condition.

In the stop state the interval timers, if already running, continue to run.

Manual Wait State

The processor enters the manual wait state and turns on the stop light and wait light when either of the following occurs:

- The System/7 power-on sequence is performed.
- The reset key on the operator console is depressed.

While in the manual wait state, no interruption requests are recognized by the processor. The processor can exit from this state when the stop, IPL, or start key on the operator console is depressed. When the start key is depressed, the processor begins program execution on the lowest priority level (level 3) at the storage location specified by the IAR.

Wait State

The processor enters the wait state and turns on the wait light whenever the level exit (PLEX) instruction is executed and no other interruption request is pending. While in the wait state, the processor will recognize interruption requests, depending on interruption controls set by the program.

The processor exits from the wait state when the stop, IPL, or reset key on the operator console is depressed, or when an interruption request requires service. Depression of the start key has no effect when the processor is in the wait state.

Load State

The processor enters the load state and turns on the load light when any of the following occurs:

- The program load key on the operator console is depressed.
- The automatic restart function is operating.
- A host processor signals the System/7 that an IPL operation is to be performed.

The processor is in the load state only during the IPL operation. (See program load (IPL) descriptions in Chapters 5, 7, 8, 9, 15 and/or "EIPL" in Chapter 16.)

PARITY

A System/7 word, whether used to hold data or an instruction, consists of 16 information bits as shown in Figure 2-3. Every 16-bit word (bits 0 to 15) contains two 8-bit bytes (bits 0 to 7 and 8 to 15). Each byte may have an odd number of bits set on (that is, having a value of 1), called odd parity. Obviously, this does not occur at all times because data or instructions can be represented with an even number of bits set on. For example, 00000000 00001001 has no bits set on in the first byte and two bits set on in the second byte. (This is an example of even parity.)

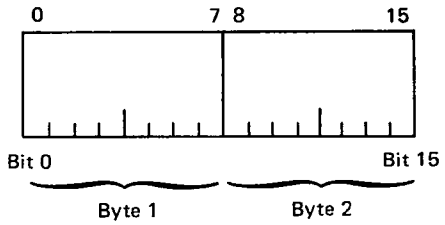


Figure 2-3. System/7 storage word

To ensure internal accuracy in data transmission and manipulation, the ability to check for odd parity is designed into the System/7 by adding two parity bits (P_0 and P_1) to each word. (So, in reality, each word in processor storage contains 18 bits.) Each parity bit is associated only with its corresponding byte as shown by the following bit organization of a System/7 word:

P_0 0 1 2 3 4 5 6 7 P_1 8 9 10 11 12 13 14 15

Each parity bit is used to maintain odd parity with its associated byte by the following method. If the 8-bit byte already has odd parity, its parity bit is set to a 0 value. If the 8-bit byte has even parity, its parity bit is set to a 1 value to give the entity (byte plus parity bit) odd parity. The previous example of even parity bytes would appear in main storage as:

P_0	P_1
1 00000000	1 00001001

Any time that the value (contents) of a word is intentionally changed during program execution, the machine automatically modifies the parity bits to maintain odd parity. Every time a word is used in the system, each byte plus its parity bit is checked to ensure that an odd number of bits is set on. If an even number of bits is detected in the processor (a parity error), a machine check error occurs. A parity error detected within the direct control channel or I/O modules is recorded in a status indicator word (for example, device status word or direct control channel status word).

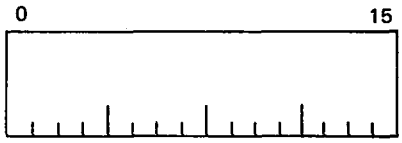
Words entering the System/7 from a host processor are analyzed to determine whether each byte has odd or even parity. Parity bits are then added to the word to maintain odd parity.

DATA FORMATS

Input/output data can be represented in a variety of ways, depending upon the medium used. (A medium is the material on which data is recorded.) For example, a paper tape that is punched with holes (which represent data) is a medium; the paper tapes can be read by a paper tape reader.

To be accessible to the program during processing, data must be in storage or a register. Thus, input data is read by an input device, placed in storage or a register, and then processed. Results (output data) are sent to an output device.

Since System/7 storage is organized in units of 16-bit words, data is read from or placed in storage on a word basis. The bits in a word are numbered left to right, from 0 to 15 as shown below.



The above format is used throughout this manual to represent a word of System/7 storage. The three major division marks inside the block are separations between hexadecimal characters in the word. The smaller divisions separate bit positions within the word.

The leftmost bit (0) is the high-order bit; the rightmost bit (15) is the low-order bit. Each bit can be set to either a binary 1 (*on* value) or a binary 0 (*off* value). Bits in a word can contain all ones, all zeros, or any combination of both zeros and ones, such as:

All zeros

Bit →	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Value →	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

All ones

Bit →	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Value →	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Any combination

Bit →	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Value →	1	0	0	0	1	1	0	1	0	1	1	1	0	0	0	1
Or →	0	1	1	1	0	0	1	1	1	1	1	0	0	0	0	0
Or →	1	1	1	1	0	0	0	0	1	1	1	1	0	0	1	1

Numeric, alphabetic, special character, or logical information can be represented by the bit values in a word, as determined by the combination of the bits in a word and the intention of the programmer who organizes the program and data in storage.

Numeric Data Formats for Arithmetic Operations

Numeric computations in the System/7 are performed in binary arithmetic by the arithmetic instructions. The value of bit 0 (the high-order bit) of a word specifies whether the word represents a positive or negative number as shown in Figure 2-4. This bit is called the sign bit. If the value of bit 0 is 0, the number represented in the word is positive; if the value of bit 0 is 1, the number represented in the word is negative.

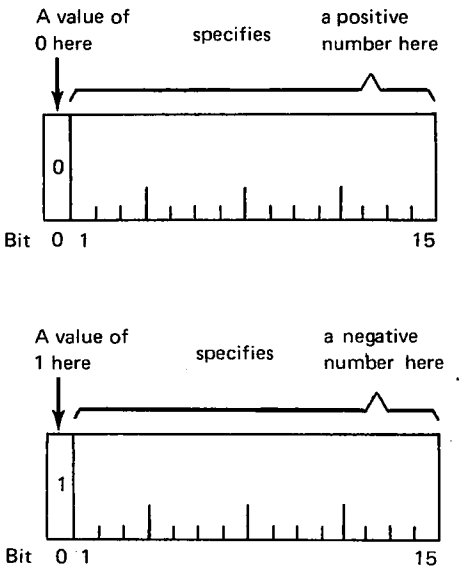


Figure 2-4. Numeric data format

The ranges of numeric values that can be represented in a 16-bit word are shown as follows:

16-bit word					
Binary					Decimal
Bit 0 (sign)	Bits 1-15 (numeric value)				
0	000	0000	0000	0000	00000
+	to				to
0	111	1111	1111	1111	32,767
1	000	0000	0000	0000	-32,768
-	to				to
1	111	1111	1111	1111	-00001

Positive numbers are always represented in true binary notation with a 0 sign bit. Negative numbers are represented in twos-complement form: the sign bit contains a 1, signifying the negative number. For example, the number 1111 1111 1111 1111 is shown above as having an equivalent decimal value of -00001. (The left 0's are shown since 16 binary bits can represent a five-digit decimal number.) This binary value is really the twos-complement of negative 1.

The absolute form can be obtained in the following manner.

1. Invert by changing each position to its opposite binary value (from all 1's to all 0's in this example):

   1111 1111 1111 1111

   becomes

   0000 0000 0000 0000

2. Add 1 to the inverted number:

   0000 0000 0000 0000

   +1

   0000 0000 0000 0001

Negative results are always represented in twos-complement form, which specifically excludes a negative 0.

STORAGE ADDRESSES

For processor module models A and B, storage capacity ranges from 2K (2,048) words to 16K (16,384) words. Storage between these limits can be obtained in 2K increments. For the model E, storage capacity ranges from 16K (16,384) words to 64K (65,536) words in increments of 4K. The address for any storage location in System/7 can be contained in a 16-bit word as in the following examples:

Binary Address (16 bits)	*Hexadecimal Address*	*Decimal Address*
0000 0000 0000 0000	0000	00000
0000 0111 1111 1111	07FF	2,047
0001 0100 1010 0000	14A0	5,280
0011 1111 1111 1111	3FFF	16,383
0100 1110 0010 0000	4E20	20,000
0111 1101 0000 0000	7D00	32,000
1111 1111 1111 1110	FFFE	65,534

REGISTERS

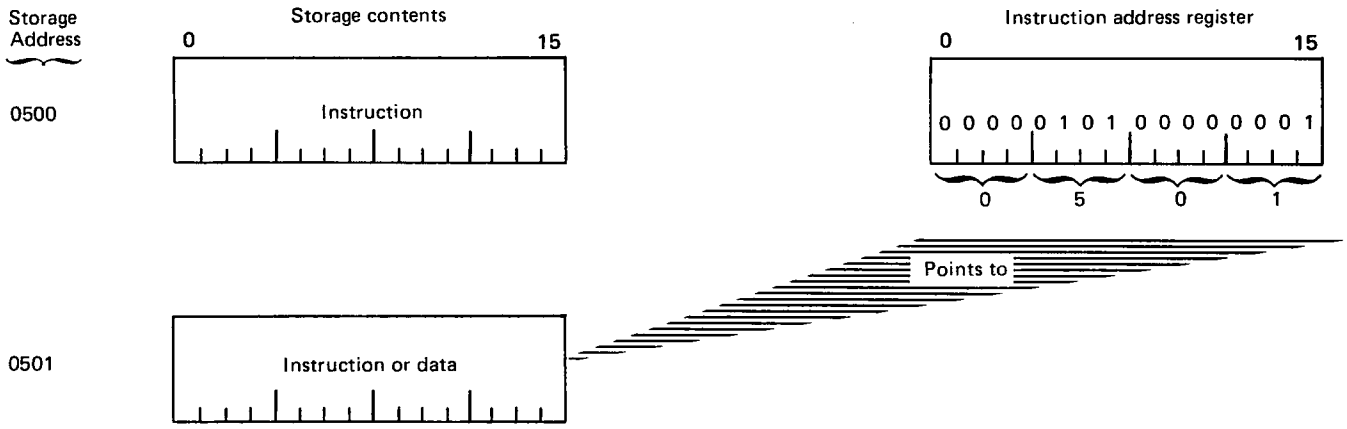
Storage Address Register (SAR)

The storage address register is a 16-bit register used by the processor to access each location in main storage. Although the contents of this register cannot be controlled directly by the programmer, the console address lights indicate the contents of the SAR during the storage access machine cycle.

Instruction Address Register (IAR)

The instruction address register (IAR) holds the address of the next instruction to be executed (see Figure 2-2). During program execution the contents of the IAR are placed in the storage address register for fetching the contents of the next location in storage. The IAR is incremented immediately after its contents have been placed in the storage address register. Consequently, the IAR contains the address of the next location in storage at the time the current instruction is executed.

For example, assume that the following instruction is being executed:



The 16-bit IAR contains the address of the storage word immediately following the instruction being executed. In most cases, this next word is the next instruction to be executed. Sometimes, however, the contents of the IAR are changed as a result of the instruction being executed. Execution of a branch instruction, for example, can cause accessing of the next instruction from a storage location other than the location immediately following the current instruction.

The System/7 processor also has three backup instruction address registers (IARB 1, IARB 2, and IARB 3) for priority interruption levels 1, 2, and 3, respectively. When a change of interruption level occurs on level 1, 2, or 3, the IAR is automatically saved in the IARB for the level that is interrupted. (There is no need for an IARB for level 0 because level 0 is the highest priority level and cannot be interrupted. Therefore, IARB 0 is used for the old IAR and will point to the address of the last instruction executed.) When a return is made to the interrupted level, processing will resume at the location specified by the saved address.

The contents of the IAR or any IARB can be displayed on the operator console. The contents of the IAR can also be modified via the console switches.

Index Registers (XR1 to XR7)

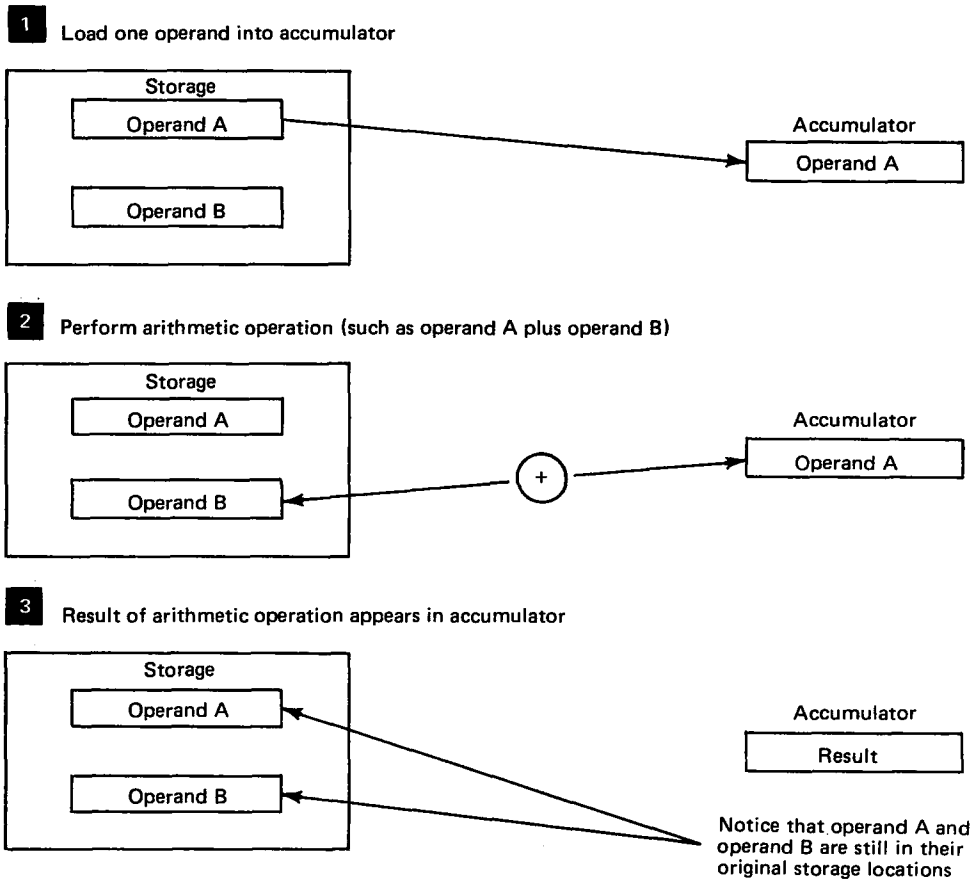
The System/7 processor has seven index registers for each of the four priority interruption levels (see Figure 2-2). Each index register has 16 data bits plus 2 parity bits. The contents of index registers can be used in generating effective addresses as described in Chapter 4 under "Effective Address Generation." The contents of index registers can also be manipulated in arithmetic and logical operations. These manipulations are explained in the detailed instruction descriptions in Chapter 4 "Processor Instructions."

The contents of any selected index register can be displayed on the operator console or modified via the console switches.

Accumulators (ACC 0 to ACC 3)

The System/7 processor has four accumulators (ACC 0 to ACC 3), one for each priority interruption level (see Figure 2-2). These 16-bit registers are used in most arithmetic and logical operations. Such operations are performed using one specified operand (from storage or from an index register) and one implied operand (the value previously loaded into the accumulator). The result is in the accumulator at the end of the operation. This result is obtained by a machine function that depends upon the instruction being executed.

The following example illustrates the use of the accumulator:



The contents of any selected accumulator can be displayed on the operator console or modified via the console switches.

Interruption Mask Register and Summary Mask

The interruption mask register is a 4-bit register shown in Figure 2-2. The bits are numbered 0 to 3, each bit corresponding to priority interruption levels 0 to 3, respectively. The appropriate bit in the mask register is set on (given a 1 value), to enable the corresponding priority interruption level. That is, interrupting sources assigned to that priority level can present interruptions to the system if their priority is higher than that of the current operating program. In addition, there is a summary mask bit which, when on, prevents acceptance of priority interruptions without changing the contents of the interruption mask register. Power/thermal warning interruptions are also disabled when the summary mask is on.

Miscellaneous Machine Registers

Storage Data Register (SDR)

The storage data register holds 16 data bits and two parity bits. Every word of data transferred to or from main storage by the processor passes through the SDR. The contents of the storage data register can be displayed by the console data lights.

Operation Register (Op)

The operation register is a 16-bit register that holds the first 16 bits of the instruction accessed from storage. In most cases these 16 bits are the complete instruction. The System/7 processor interprets these bits and performs the operation required. The contents of the operation register can be displayed by the data lights on the operator console.

PROGRAM INDICATORS

The System/7 has six indicators that show the status of the result field after an instruction is executed. These indicators are called carry, overflow, result-zero, result-even, result-positive, and result-negative. At the completion of most instructions, these program indicators are set accordingly. Some of the branching instructions can test these indicators for conditional branching situations.

Once an instruction turns the indicators either on or off, they remain in this state (and can be tested) until the execution of another instruction that changes the indicators according to the result of the operation. (Refer to the detailed description of the individual instructions in Chapter 4, "Processor Instructions," to determine the effect of each instruction on the setting of these six program indicators.)

Each of the four priority interruption levels has a set of program indicators. The carry and overflow indicator settings are described separately in each instruction description. The four result indicators are grouped together and referred to as the result indicators, since they always indicate the results of an operation in a register or in the accumulator.

Carry Indicator

The carry indicator is used for three functions:

1. To indicate that an operation has produced a result that exceeds the physical capacity of the system.
2. To hold one of the two condition code bits during an I/O instruction.
3. To hold the summary status bit during interruption servicing.

The following three conditions turn on the carry indicator when the physical capacity of the accumulator is exceeded:

1. During a shift-left operation (excluding shift-left circular) when the last bit shifted out of bit position 0 (the sign bit) was a 1-bit.
2. During an add operation, when a carry out of bit position 0 (the sign bit) was a 1-bit.
3. During a subtract operation, when a borrow beyond bit 0 occurs.

The carry indicator is automatically reset prior to each add, subtract, and noncircular shift-left operation.

At the conclusion of an I/O instruction the carry and overflow indicators are set to reflect a two-bit condition code returned by the I/O device. Values of the condition code are covered in more detail in Chapter 4 under "Input/Output Instruction."

The use of the carry indicator to hold the summary status bit during interruptions is covered more fully in Chapter 3, "System/7 Interruptions."

Overflow Indicator

The overflow indicator is set on if the result of an arithmetic operation exceeds the logical capacity of the accumulator. This happens when an add or subtract operation produced a result greater than $2^{15}-1$ or less than -2^{15}. This range is from +32,767 to −32,768 in decimal notation, or from 7FFF to 8000 in hexadecimal notation.

The overflow indicator may be reset only when it is tested by either the skip or branch conditional instruction with the overflow-save flag equal to 0. These two instructions are described in more detail in Chapter 4, "Processor Instructions." The overflow indicator is also used as part of the I/O condition code as described previously under "Carry Indicator," and in Chapter 4 under "Input/Output Instruction."

Result-Zero Indicator

The result-zero indicator is set on when the result field of an operation contains all 0's.

Result-Even Indicator

The result-even indicator is set on when the low-order bit (bit 15) of a result field is set to 0, indicating that the operation has produced an even number.

Result-Positive Indicator

The result-positive indicator is set on when the high-order bit (bit 0, the sign bit) of a result field is set to 0, indicating a positive number. If the numeric value of the entire result field is 0, this indicator is set off because the result-positive definition does not include a 0 value.

Result-Negative Indicator

The result-negative indicator is set on when the high-order bit (bit 0, the sign bit) of a result field is set to 1, indicating a negative number. A negative 0 is not represented in the System/7.

STORAGE PROTECT (5010 PROCESSOR MODULE MODEL E ONLY)

Storage protect on the 5010 Model E provides protection against inadvertent storing in a selected location. No read protection is provided; this protection is only against storing.

Storage is divided into blocks of 512 words. Each block has a storage key 3 bits long set by the alter storage key instruction. (See "Storage Protect Instructions" in Chapter 4).

Each of the 4 priority interruption levels is provided with a 3-bit protect key. A protect key can be assigned in 2 ways. The first method is by an alter protect key instruction.

The second method is automatic and occurs during the first instruction read cycle following an interruption. The protect key register for the new level is set to the same value as the storage key associated with the block containing the next instruction to be executed. This provides automatic assigning of the correct key following an interruption. Validity of the key picked by this mechanism is ensured by the location of the start instruction address in the vector table in that area of storage controlled by the program. In this case, the address arrived at by the automatic sublevel branching technique is in a storage block with the correct key assigned to this particular interruption routine.

This method of control of the storage protect mechanism permits using a storage load that does not recognize the existence of storage protection provided that a system reset has disabled it before execution of the storage load commences. The system does not support a mixed storage load in which some programs use the storage protect mechanism and others do not. In this case, an attempt by any program to access a protected area without a key match causes a program check interruption.

The storage protect mechanism is disabled (turned off) by power-on reset, system reset, the reset caused by check restart, initial program load, the occurrence of a class interruption, and the execution of an SVC instruction. The storage and protect keys are not changed by any of these actions. The keys are changed only by the appropriate alter storage or protect key instructions.

The storage protect mechanism is enabled (turned on) by the execution of an alter storage key or an alter protect key instruction. If these instructions are executed when storage protect is already enabled, it remains enabled.

The protect key is read using the read protect key instruction. If any of the storage protect instructions are issued on the 5010 Processor Module models A or B, a program check condition results and the instruction is not executed.

To write into main storage, at least one of the following conditions must be true:

1. The storage key of the addressed block must be 111.
2. The protect key currently in use must be 000.
3. The storage key of the addressed block must match the current protect key.

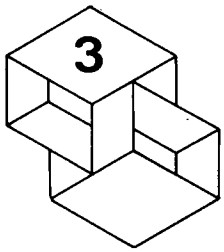
The first two conditions above indicate that a storage key of 111 permits writing into any storage location in that block regardless of the protect key used, and that a protect key of 000 authorizes writing into any location in storage regardless of the storage key value of the addressed block.

If none of the three conditions specified above is true, the attempt to write is prevented, the contents of main storage are not changed and a program check interruption occurs with invalid storage address set in the processor status word.

For certain hardware functions that involve accessing main storage, the key comparison mechanism is suppressed. For these cases, no storage protect checking is performed until the hardware function is completed. Because of this:

1. No storage protection is provided during an initial program load, since it is preceded by a hardware reset, and no instructions are executed until the initial program load is complete.
2. No storage protect checking is performed while the system is in the stop state and store operations are being performed from the System/7 console.
3. No storage protect checking is provided during a cycle steal operation.
4. The hardware controlled storing of old instruction addresses during class interruptions is not affected by storage protect.

CHAPTER CONTENTS



System/7 configurations can include a large number of devices and attachments capable of sending and receiving data. Such a device or attachment can be an I/O module or an internal machine checking circuit. Some operations are more important than others, so the System/7 program must be able to recognize the more important requests for processing in the job being performed. Such requests are presented from some external source to the processor as interruptions. That is, the current operation being performed by the processor is stopped, temporarily, so that a more important operation can be processed. The status of the interrupted operation is saved in the machine registers, the higher-priority operation is performed, and then control of the system is returned to the interrupted operation in the program.

Two types of interruptions can occur on the System/7—priority interruptions and class interruptions. Each type is explained further in the following sections.

PRIORITY INTERRUPTIONS

Priority Levels

The priority interruption scheme establishes four priority levels (numbered 0 to 3) for interrupting sources to obtain servicing by the processor. Each priority level has 16 sublevels, and each sublevel can be assigned to a different processing routine, if desired. As long as no higher priority device requests attention, current program processing and communication with I/O devices continues to completion. Upon completion of the current processing, the program must exit from the level. If no interruptions are pending at this time, the machine goes into the wait state. In the wait state no processing is performed, but the machine can recognize and accept interruptions that are expected to occur.

If interruptions are pending on the same or lower priority levels, when the current program level exit occurs, the machine does not enter the wait state, but immediately begins servicing the highest pending priority interruption. The machine will execute one instruction on the highest level in the backup register before accepting the new interrupt. Current processing is interrupted only by requests on a higher priority level, not for requests on a higher sublevel within the current priority level. For example, if the machine is processing sublevel 9 on priority level 2, an interruption on sublevel 5, priority level 2 will not interrupt the processing for sublevel 9.

If an interruption occurs that is assigned to a higher priority level than the currently executing program, the current program's status is automatically saved in the machine registers assigned to the level that was interrupted, and processing transfers to the routine that services the higher priority request. This routine can, in turn, be interrupted by a still higher priority request. The original program is not resumed until all higher priority interruptions are serviced. Figure 3-1 illustrates the priority interruption scheme just discussed.

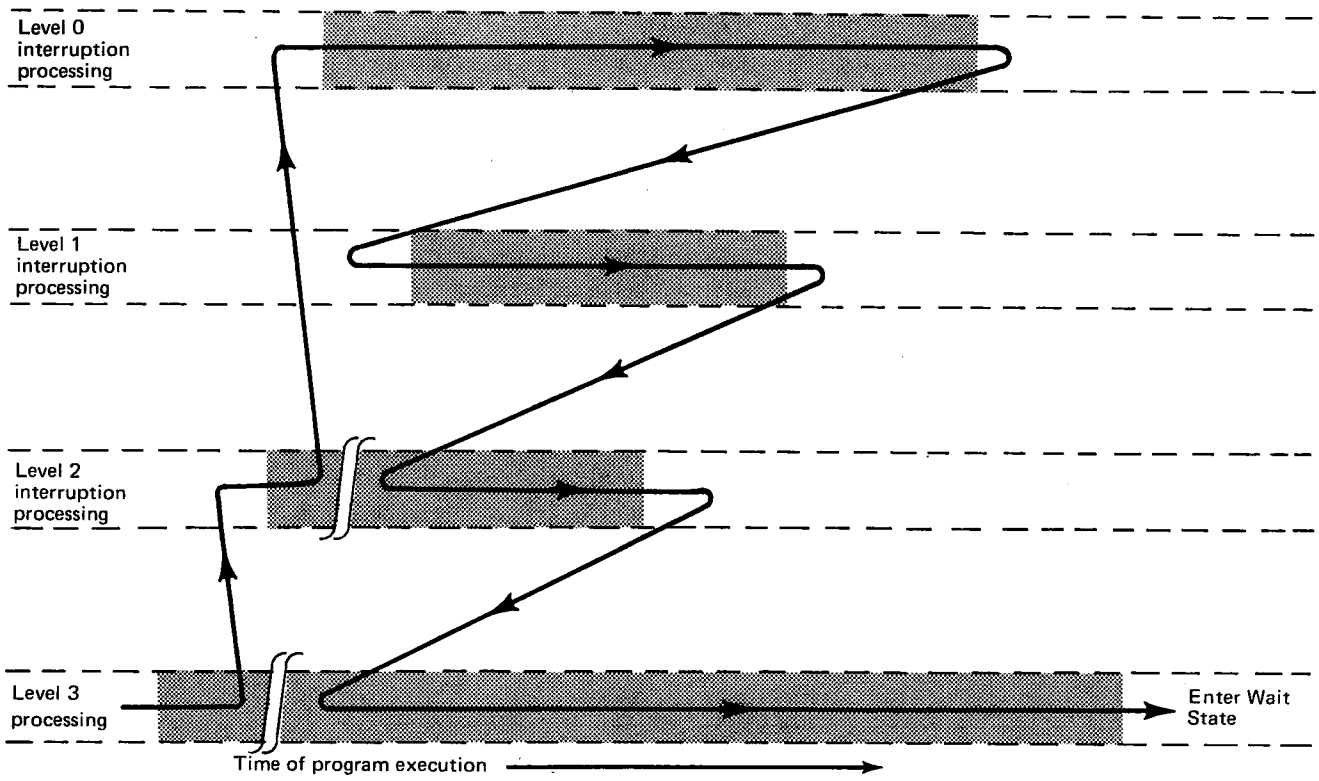


Figure 3-1. Priority interrupt system operation

Level Assignments

Interruption priority levels and sublevels can be assigned to I/O devices via program control so that the priority organization can be restructured as the application changes.

Before an I/O device can request an interruption to the System/7 processor, the I/O device must be prepared by a special I/O instruction. The instruction is covered in more detail in Chapter 4, "Processor Instructions." However, a general description of the instruction is necessary at this time in order to describe the System/7 interruption scheme more fully.

The prepare I/O instruction addresses an I/O device and "tells" the device:

1. If the device can interrupt.
2. What priority level to use for interruptions.
3. What sublevel within the priority level the device should return for processing the interruption.

Level Switching

The System/7 processor can accept interruptions after execution of any instruction prior to fetching the next sequential instruction, or when the processor is in the wait state. (See "Processor States" in Chapter 2.)

Program switching can occur from one level to another with minimum overhead because the machine status of the interrupted program does not have to be saved by programming. Instead, each of the four priority levels in the processor has its own set of registers (an accumulator, a set of six program indicators, an instruction address register, and a set of seven index registers). Thus, machine status switching following an interruption in System/7 is performed in less than 1 microsecond.

Interruption Request Stacking

Interruptions from devices are not serviced immediately if processing is taking place at the same, or at a higher, priority level. The system stacks, or queues, such pending interruptions for servicing when possible. (See Figure 3-2.)

Each of the four interruption levels has one interruption buffer in the processor module. The first request for any particular level is placed in its corresponding buffer. When the processor is able to service the request in the buffer (that is, no higher-priority interruptions are pending and the processor has completed execution of an instruction), the request is removed from the buffer and serviced. If another request is pending at that level, it enters the buffer where it is available for servicing when the present program is completed and/or system control is returned to that level.

When the buffer is full (occupied by a request not yet honored), another device assigned to the same interruption level requesting service cannot be recognized. This interruption request is stacked by the I/O module or device and must compete with any other interruption requests assigned to that level when the buffer becomes available again.

No matter how many interruptions are pending from devices on a particular priority level, the next one to be serviced when control returns to that level is the request in the buffer.

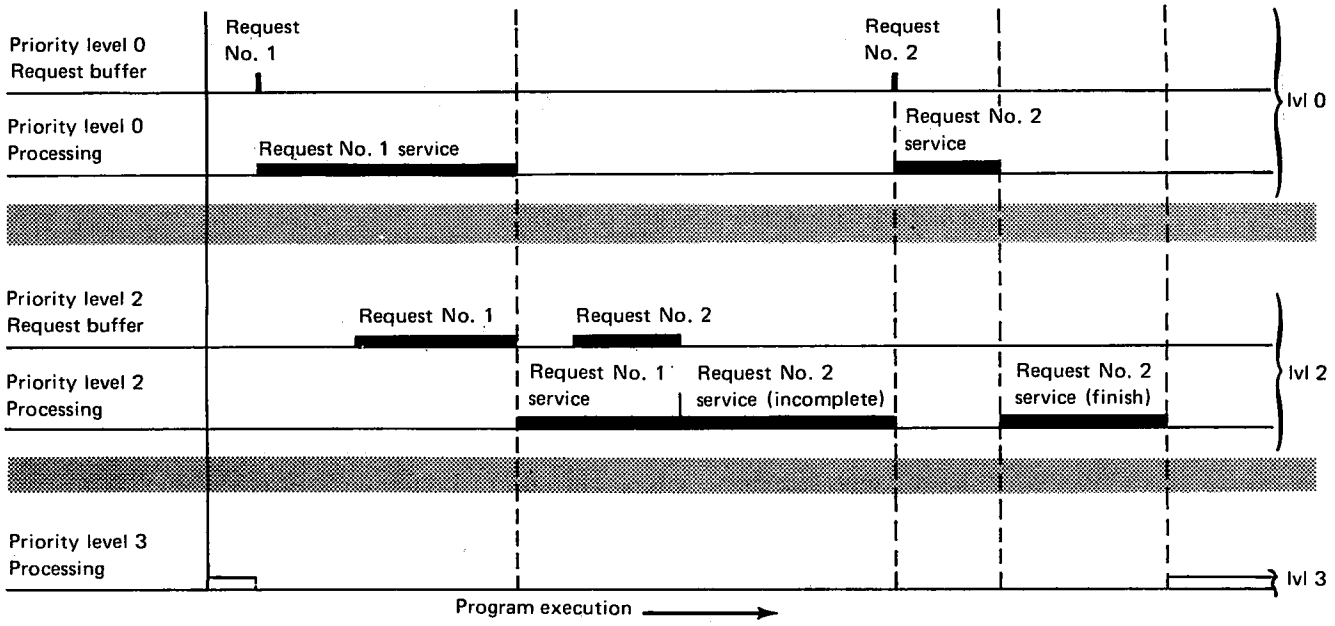


Figure 3-2. Interruption request stacking

Interruption Mask Register and Summary Mask

To give the user some control over the automatic interruption system, a 4-bit interruption mask register is provided. Bit positions 0 to 3 in the mask register correspond to priority levels 0 to 3, respectively.

If the appropriate bit in the mask register is set on, devices assigned to that priority level can interrupt the system. Setting the bit off disables the level so that no interruptions are honored on that level. The interruption mask register can be used to further modify the sequence in which the priority interruptions are serviced by enabling or disabling interruptions on one or more priority levels.

The contents of the mask register can be modified or examined by three instructions; AND to mask, OR to mask, and sense level and mask, as explained in Chapter 4, "Processor Instructions."

For additional control there is a summary mask function that, when turned on, inhibits *all four* priority level interruption *and* also power/thermal warning interruptions. Turning on the summary mask function does not disturb the contents of the interruption mask register; the summary mask simply has precedence over the contents.

The summary mask function is turned on by any of the following:

1. Executing the supervisor call instruction
2. Executing the sense level and mask instruction with bit 11 of the instruction on
3. A machine check or program check class interruption
4. A power/thermal warning class interruption that occurs when the summary mask is off.

The summary mask function is turned off by any of the following:

1. Executing the AND to mask instruction. (If the contents of the interruption mask register are not to be disturbed, the register specified by the instruction must have bits 0–3 set to 1111.)
2. Executing the OR to mask instruction. (If the contents of the interruption mask register are not to be disturbed, the register specified by the instruction must have bits 0–3 set to 0000.)
3. Executing the branch and unmask long instruction (processor module Model E only).
4. A system reset or power-on reset.

Reserved Storage Locations

An area in main storage is reserved for:

1. The restart instruction
2. Address of 5024 IOTB
3. Address required to service class interruptions
4. Address pointers to level tables

These storage locations and their contents are shown in Figure 3-3. Hex locations 2 through 13 may contain data other than that shown in this figure during an initial program load operation. However, locations that are not used by the executing program may be used during program operation.

Storage location (hexadecimal)		Contents
	0	Restart instruction
	1	
	2	Reserved
	3	
	4	
	5	
	6	
	7	
	8	Address of 5024 IOTB
Class vectors	9	Program-check start instruction address and branch address for supervisor call instruction
	A	Power failure start instruction address
	B	Machine-check start instruction address
	C	Power failure old instruction address
	D	Machine-check old instruction address
	E	Reserved
	F	
Level vectors	10	Address of level 0 displacement table
	11	Address of level 1 displacement table
	12	Address of level 2 displacement table
	13	Address of level 3 displacement table

Figure 3-3. Reserved storage locations

The reserved storage locations are described as follows:

Hex Storage Location	*Contents*
0–1	The restart instruction. After loading the IPL routine, the machine automatically branches to location 0 to initiate processing. This instruction is also used with the check control switch in the check-restart position as described in Chapter 5, "Operator Console."
2–7	Reserved.
8	Address of 5024 IOTB (refer to Chapter 17, "IBM 5024 I/O Attachment Enclosure" for a further description of the 5024 IOTB).
9–B	The addresses of the first instructions in the routines that service program check, power failure, and machine check interruptions, respectively.
C–D	The IAR backup contents after a power failure or machine check interruption, respectively. This is the old instruction address (OIA). Control can then be returned to the correct instruction in the interrupted program after successful completion of the interruption servicing routine. Only one word is required to save the interrupted program address for each type of interruption because only one power failure or machine check can occur at any one time (it is not possible to have a machine check or power failure at more than one interruption level at the same time).
E–F	Reserved.
10–13	The addresses of the first word in the level displacement tables for interruption levels 0 to 3, respectively. Program-check old instruction address is stored at displacement 0 in the displacement table of the level where the program-check occurred. See Figure 3-4.

Locations 0 to 13 are fixed as to size and function of contents. However, additional storage is required for the level displacement tables assigned to each of the priority interruption levels (0, 1, 2, and 3). These tables are variable both in number (from 1 to 4) and individual size (from 2 to 17 words). They can be located anywhere in main storage except the reserved locations discussed above. Figure 3-4 shows the format for a priority level table.

The storage space required for the level displacement tables depends on two things:

1. The number of different priority levels assigned to I/O devices by the user.
2. The number of sublevels assigned to I/O devices on the same priority level.

For example, if only two different priority levels are used (such as levels 0 and 1), only two level displacement tables are required in storage. In addition, if sublevels 0 through 5 are assigned to devices that are on priority level 0, and sublevels 0 through 11 are assigned to devices that are on priority level 1, then only seven words would be required for the level 0 displacement table and 13 words for the level 1 displacement table.

Each priority level displacement table must occupy a contiguous block of storage, but need be only as long as required by the current assignment of sublevels. If more than one priority level is used, the level tables do not have to be adjacent to each other.

Table word (hexadecimal)		Contents (hexadecimal)
	n	Level 0 program - check old instruction address
	n + 1	Level 0 sublevel 0 interrupt routine address
	n + 2	Level 0 sublevel 1 interrupt routine address
	n + 3	Level 0 sublevel 2 interrupt routine address
	n + 4	Level 0 sublevel 3 interrupt routine address
	n + 5	Level 0 sublevel 4 interrupt routine address
	n + 6	Level 0 sublevel 5 interrupt routine address
Ascending	n + 7	Level 0 sublevel 6 interrupt routine address
storage	n + 8	Level 0 sublevel 7 interrupt routine address
locations	n + 9	Level 0 sublevel 8 interrupt routine address
	n + A	Level 0 sublevel 9 interrupt routine address
	n + B	Level 0 sublevel A interrupt routine address
	n + C	Level 0 sublevel B interrupt routine address
	n + D	Level 0 sublevel C interrupt routine address
	n + E	Level 0 sublevel D interrupt routine address
	n + F	Level 0 sublevel E interrupt routine address
	n + 10	Level 0 sublevel F interrupt routine address

Figure 3-4. Priority level table format

Interrupt Sublevel Branching

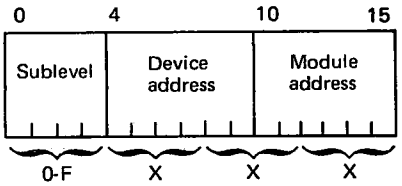
The user assigns a priority level and sublevel to an I/O device by a single prepare I/O command. Each I/O device that is to request interruptions to the processor must be prepared by a separate prepare I/O command, which assigns a priority level and sublevel to the device.

Upon detecting an interruption (from a source prepared for interruptions), the system examines the priority level to determine whether the interruption is to be permitted according to the interruption mask register. If the same level or a higher level is processing and the interruption is permitted, the request remains in the buffer. If the same level or a higher level is not processing and the interruption is permitted, the interruption request is honored.

Using the level number, the system accesses the branch table reserved for that level. Using the sublevel as a displacement value, the system locates the desired address in the branch table and then branches to that address. This address is the starting address of the servicing routine that the user has established in main storage for the interrupting source. This total interruption process is performed in less than 1 microsecond.

Multiple interrupting sources (I/O devices) can be assigned to the same interruption level and sublevel. Thus, a common interruption servicing routine can be used for multiple or similar I/O devices. The one device (out of two or more assigned to the same interruption level and sublevel) that actually caused the interruption is determined by the program's examining the accumulator contents immediately after the interruption.

The accumulator associated with the level contains the address of the I/O device causing the interruption, in the following format:



During interruption presentation, an I/O device also presents a summary status bit (S), which is placed in the carry indicator. The carry indicator is set to 0 if a normal interruption condition occurred, or to 1 if an exception interruption condition occurred. Exception conditions that set the summary status bit are noted in descriptions of I/O commands for I/O devices, later in this manual.

CLASS INTERRUPTIONS

Internal machine error conditions can cause three types of class interruptions:

1. Machine check, caused by a machine error or by program execution.
2. Program check, caused by an invalid machine instruction.
3. Power/thermal warning, caused by a power or thermal irregularity.

Class interruptions alert the system to error conditions which may prevent further processing. Machine checks and program checks cannot be prevented from interrupting the system. Power/thermal warning class interruptions, however, are controlled by the summary mask and are inhibited during execution of the supervisor call (SVC) instruction. (Supervisor call is discussed in more detail in Chapter 4.)

Because of their nature, class interruptions immediately disable all four priority interruption levels and cause a branch via one of the main storage locations reserved for class interruption start addresses. (See "Reserved Storage Locations" in this chapter.) Machine checks and program checks turn on the summary mask function. A power/thermal warning that occurs when the summary mask function is off will also turn it on.

A class interruption does not cause a change in priority level; the interruption is serviced on the level that is active when the error condition occurs. If no level is active when a power/thermal warning interruption occurs, the interrupt is serviced on priority level 3. When the class interruption is serviced, the servicing program should save any register contents and/or status information needed to restart the program.

Lights on the operator console indicate the cause of a class interruption when the processor is in the stop state. In addition, a machine instruction (load processor status) can check the processor status to determine the specific cause of the interruption. This instruction is described in more detail in Chapter 4.

Priority of Class Interruptions

Although class interruptions are serviced on the current priority level, they are serviced according to class priority hierarchy. A machine check has the highest priority, a program check has the next priority, and a power/thermal warning has the lowest priority. The load processor status instruction resets all class interruption indicators except for power/thermal warnings.

Machine Check

Machine check interruptions occur when processor error checking circuits detect a machine malfunction. Malfunctions can be caused by:

1. A parity error in an index register.
2. A parity error encountered by the processor attempting to access main storage.
3. More than one type of machine cycle requested or taken simultaneously.
4. More than one interruption level being executed simultaneously, or no level is active and the processor is taking machine cycles.
5. An I/O hardware error occurring on the internal interface.

Note: A machine check error can result from programming if a storage location or an index register is used by the program and the location or register was not loaded with an initial value after the system was powered on. A machine check can also result from a software error when a program check is encountered between the execution of a supervisor call instruction and a load processor status instruction, or when another supervisor call instruction is issued.

Operator console lights indicate the cause of a machine check when the processor is in the stop state. Only one machine check can occur at any one time in the system.

When a machine check interruption occurs, all I/O devices and pending interruption requests are reset. The address of the instruction being executed when the error occurred is saved automatically in storage location hex D. The contents of storage location hex B, which contains the starting address of the machine check servicing routine, are then automatically loaded into the IAR and the servicing routine is given control. (Refer to "Reserved Storage Locations" in this chapter.)

Program Check

Program check interruptions occur when the processor detects a machine instruction that is invalid for one of the following reasons:

1. Invalid operation code (an invalid instruction for the System/7 processor).
2. Invalid modifier field contents in instructions with operation code of 11111. (On processor module model E, this condition causes a no-operation and does not result in a program check.)
3. Invalid address (a main-storage address that exceeds the limits of the storage installed in the system).
4. Invalid function field (000) in an execute I/O instruction (processor module models A and B only).
5. Invalid shift count value in shift instructions.
6. A cycle steal I/O instruction has been issued and there is no cycle steal feature installed in the processor.
7. The processor has attempted to alter a storage protected area without the proper key (processor module model E only).

When the machine is in the stop state, operator console lights indicate the cause of a program check.

When a program check interruption occurs on one of the four priority interruption levels, the address of the instruction being executed when the program check was detected is saved automatically in the first word of the appropriate level table. The program check may be due to an invalid instruction, or an invalid address generated by an instruction. If an invalid instruction is detected, the first word of the appropriate level table (OIA) contains the instruction address. If an invalid address is detected, the OIA normally contains the address of the instruction that generated the invalid address. (There is one exception to this. If a higher level priority interruption occurs during a branch to an invalid address, and the interrupt is accepted immediately following the branch instruction, then the invalid address is not used or checked until control is returned to the lower priority level. Under these circumstances, the OIA contains the address of the PLEX instruction that released control from the higher priority level.) The contents of storage location 9, which contains the starting address of the program check servicing routine, are then automatically loaded into the IAR and the servicing routine is given control. (Refer to "Reserved Storage Locations" in this chapter.)

Power/Thermal Warning

A power/thermal warning interruption occurs when the system senses a condition caused by voltage or temperature changes that border on the operating limits of the system. (Refer to "Power Failure and Thermal Warning" in Chapter 1.) Power/thermal warning class interruptions are controlled by the summary mask. If the summary mask is on, the class interruption is inhibited.

Power/thermal warning interruptions are also affected by the settings of the console switches as shown in Figure 3-5.

When a voltage or temperature condition causes a power/thermal warning interruption, the IAR contents (address of the next instruction in the interrupted program) are saved automatically in storage location hex C. The contents of storage location hex A, which contains the starting address of the power/thermal warning servicing routine, are then automatically loaded into the IAR and the servicing routine is given control. (Refer to "Reserved Storage Locations" in this chapter.)

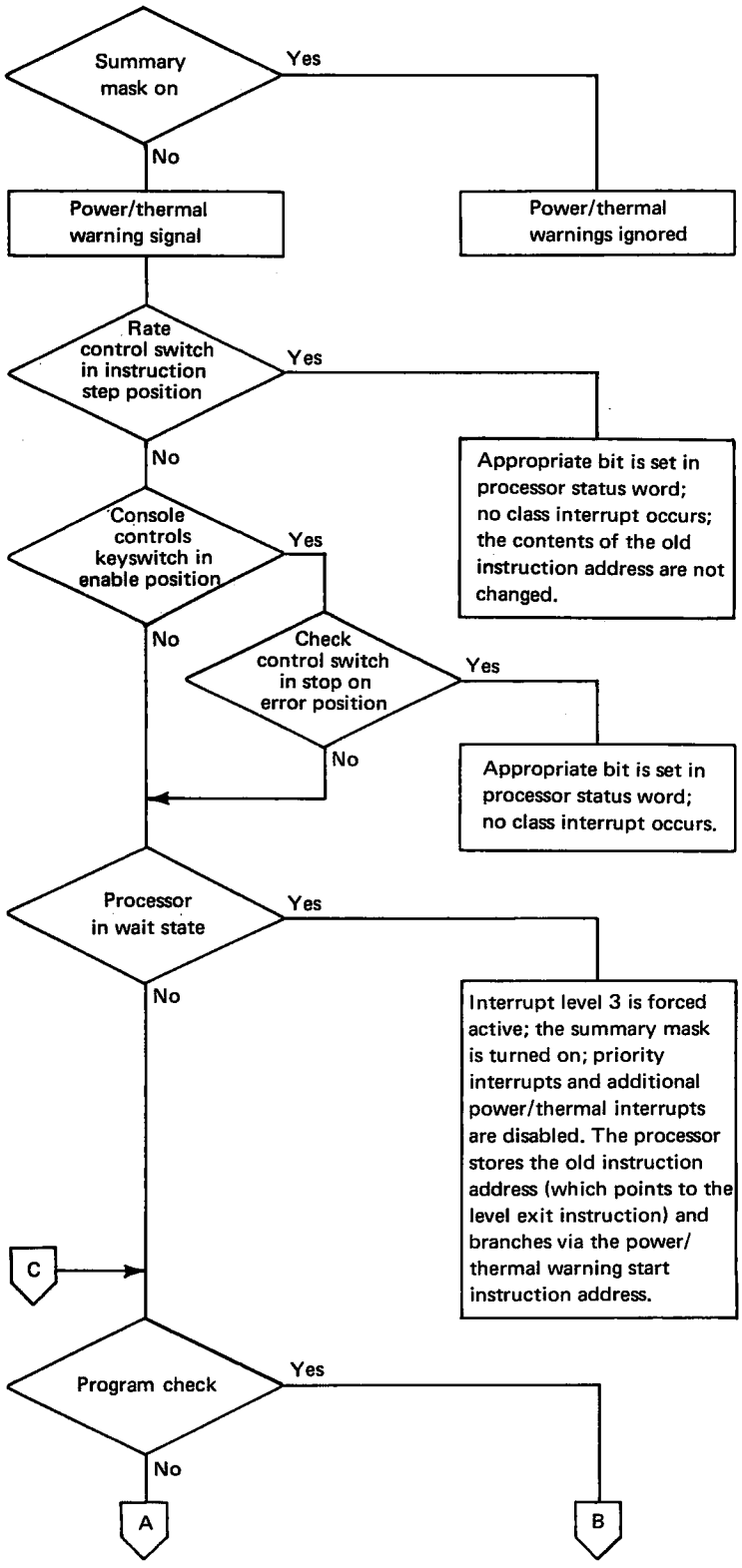


Figure 3-5. Console switch settings for power/thermal warning interruptions (sheet 1 of 2)

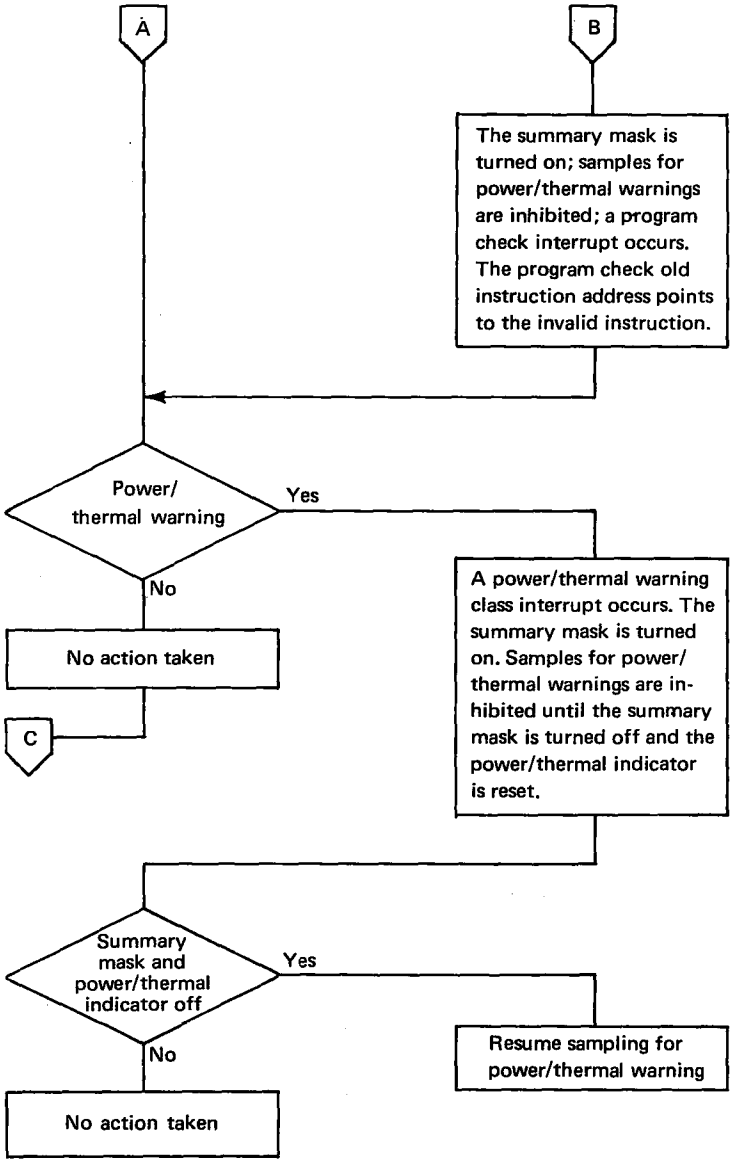


Figure 3-5. Console switch settings for power/thermal warning interruptions (sheet 2 of 2)

Power Warning

System/7 circuits are protected from adverse power conditions by sensing circuits that check the input line voltage in each 5026 Enclosure. When the input power falls below 85 percent of nominal voltage, the sensing circuits alert the processor module with a power/thermal warning interruption request as shown in Figure 3-6. The time between warning and shutdown depends on the system load and the type of input power failure. This time is a minimum of eight milliseconds with an instantaneous primary power shutdown, for a System/7 containing close to the maximum possible number of I/O modules.

Power shutdown to an enclosure occurs when the primary power falls to approximately 60 percent of its rated value. If input power remains below 85 percent but does not fall enough to completely shut down the power system, the power warning status indicator to the processor module remains active.

Note: The power warning feature just described is an optional feature in System/7 enclosures. However, loss of any one of the several different voltages used in the system will remove power from the entire enclosure in order to protect the electronic circuits.

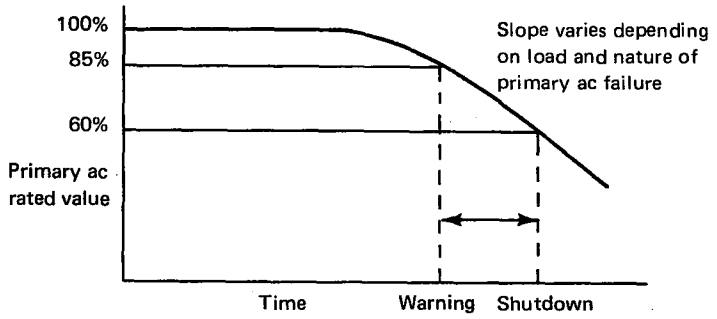


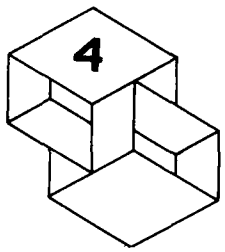
Figure 3-6. Power failure warning

Thermal Warning

In order to protect the electronic components within the system, each module in a 5026 Enclosure contains a temperature sensing mechanism. Should the operating temperature in a module approach an excessively high value the sensing mechanism alerts the processor module with a power/thermal warning interruption request. A status indicator to the processor can be checked by the System/7 program to determine if the master unit or a remote unit caused the warning. If the over-temperature condition exists for 1 to 6 seconds, the sensing mechanism shuts off power to that enclosure. The over-temperature condition is also indicated by the thermal light on the operator console. When the thermal light goes off, power can be restored by turning the power switch to off and then to on.

A thermal power shutdown in the master unit shuts off power to *all* enclosures configured in the system. However, a power shutdown in a remote unit shuts off power to the remote unit only.

CHAPTER CONTENTS



The System/7 processor fetches and executes instructions in order to perform operations on data (such as testing, arithmetic, movement, and logical decision operations). Instructions used by the processor are represented in a storage word by fixed patterns of bits, each pattern indicating a specific operation to be performed by the processor in a certain predetermined manner.

INSTRUCTION CLASSES

For purposes of discussion and clarification, the instruction set is divided into classes which describe, in general, the type of data operation that is performed.

Mnemonic symbols are used to abbreviate the instruction names when programming in Assembler language. Use of the leading P in the instruction mnemonics is required in all System/7 programs to be assembled on an 1130 or 1800 system, or using the System/7 Stand-Alone assembler. The P or an X is always required in the Execute I/O instruction (PIO or XIO). The P is optional (except for PIO) when assembling on a System/360, System/370, or the System/7 Macro Assembler. The System/7 instructions, along with their mnemonics, operation codes, and execution times, are shown in Figure 4-1.

Instruction	Mnemonic[1]	Operation code	Execution time in nanoseconds
Load and store			
Load accumulator	[P]L	11000	800
Load and zero	[P]LZ	11001	1200
Load immediate	[P]LI	01100	400
Load index long	[P]LXL	10001	1200
Store accumulator	[P]ST	11010	800
Store index	[P]STX	01101	800
Arithmetic			
Add	[P]A	10000	800
Subtract	[P]S	10010	800
Add register	[P]AR	11111	400
Subtract register	[P]SR	11111	400
Complement register	[P]CR	11111	400
Add immediate	[P]AI	01110	400
Logical			
AND	[P]N	11100	800
OR	[P]O	11101	800
Exclusive OR	[P]X	11110	800
AND register	[P]NR	11111	400
OR register	[P]OR	11111	400
Exclusive OR register	[P]XR	11111	400
Shifting			
Shift left logical	[P]SLL	00010	400 + 50N + 50 if odd no. shifts
Shift left circular	[P]SLC	00010	400 + 50N + 50 if odd no. shifts
Shift right logical	[P]SRL	00010	400 + 50N + 50 if odd no. shifts
Shift right arithmetic	[P]SRA	00010	400 + 50N + 50 if odd no. shifts
(N is the number of bits shifted)			
Branching			
Branch	[P]B	00111	400
Branch and link	[P]BAL	01011	400
Branch and link long	[P]BALL	01010	800
Branch and unmask long[2] [3]	BUL	01010	800
Branch conditional	[P]BC	01000	400—no branch 800—branch taken
Skip conditional	[P]SKC	01001	400
Add to storage and skip	[P]AS	01111	1200
Register-to-register			
Store indicators[2] [3]	STI	11111	400
Store to register	[P]STR	11111	400
Load from register	[P]LR	11111	400
Interchange register	[P]IR	11111	400
Load processor status	[P]LPS	11111	400
Inspect IAR backup	[P]IIB	11111	800
Read IAR backup[2] [3]	RIB	11111	800
Write IAR backup[2] [3]	WIB	11111	800
AND to mask	[P]NM	10110	400
OR to mask	[P]OM	10110	400
Sense level and mask	[P]SLM	11111	400

Figure 4-1. (Part 1 of 2) System/7 Instructions

Instruction	Mnemonic[1]	Operation code	Execution time in nanoseconds
State control			
Level exit	[P] LEX	00110	400
Stop	[P] STP	00100	400
Supervisor call[3]	SVC	10011	⩽2000
Input/output			
Execute I/O	PIO[5]	00001	1800 + D[6]
Alter storage key[2] [3]	ASK	00001	1200–1800[4]
Alter protect key[2] [3]	APK	00001	1200–1800[4]
Read protect key[2] [3]	RPK	00001	1200–1800[4]

Notes.

1. Use of the letter P is required in all programs assembled on an 1130 or 1800 system, or using the standalone assembler. The P is optional for other use except for the PREP instruction, where it is required.
2. Available only with model Exx.
3. These instructions are not supported by the System/7 Standalone Assembler or for System/7 programs assembled on an 1130 or 1800 system.
4. The system treats these instructions as I/O instructions. Therefore, interface delay times cause the range of execution times.
5. An X may be substituted for the P in this instruction.
6. D, the delay inherent in a system configuration, varies from 100 to 2100 ns depending on the physical location of the I/O module concerned. D varies from 100 to 800 ns on systems without a 5026 Enclosure Model D3 or D6.

Figure 4-1. (Part 2 of 2) System/7 Instructions

INSTRUCTION FORMATS

Two instruction formats are used in the System/7 processor: a 16-bit short-format instruction (one word) and a 32-bit long-format instruction (two words), as shown in Figure 4-2.

Both instruction formats are divided into several fields whose contents specify the operation to be performed and the location of the data to be processed. Most format fields have the same meaning from one instruction to the next; some, however, have special meanings, depending on the particular instruction. These exceptions are discussed in the detailed description of the instruction.

The operation-code (Op code) field is five bits in length and specifies the instruction to be performed.

The register (R) field is three bits in length and indicates whether the accumulator, instruction address register, or one of the index registers is to take part in the instruction execution. An R field of 000 specifies either the accumulator or the instruction address register (IAR), depending on the instruction. The index registers are specified by the following R fields:

001 – index register 1
010 – index register 2
011 – index register 3
100 – index register 4
101 – index register 5
110 – index register 6
111 – index register 7

The contents of the R field register are used for (1) generating an effective address to locate the operand, (2) as an instruction operand, or (3) as data sent to or received from an I/O device.

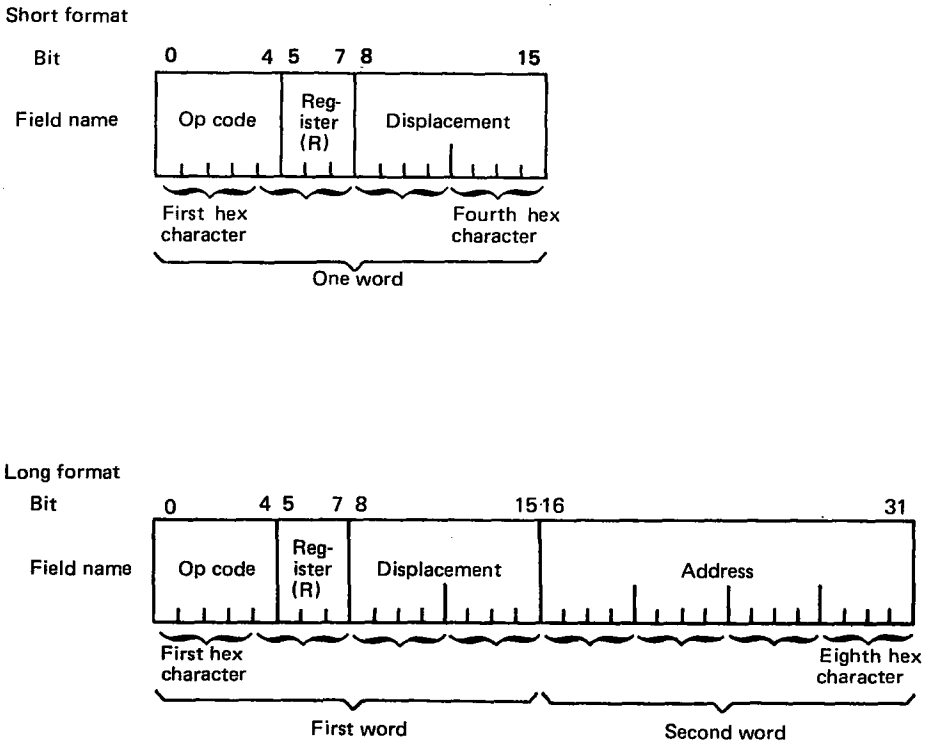


Figure 4-2. Instruction formats

The displacement field is eight bits long. Its function is determined by the instruction to be performed, as indicated in the operation-code field. The displacement field may contain:

1. Data to be used for generating an effective address (EA) for the instruction being performed. (See "Effective Address Generation," which follows.)
2. Data to be manipulated in some manner by the instruction.
3. Data that modifies the operation code so that a particular instruction is executed.

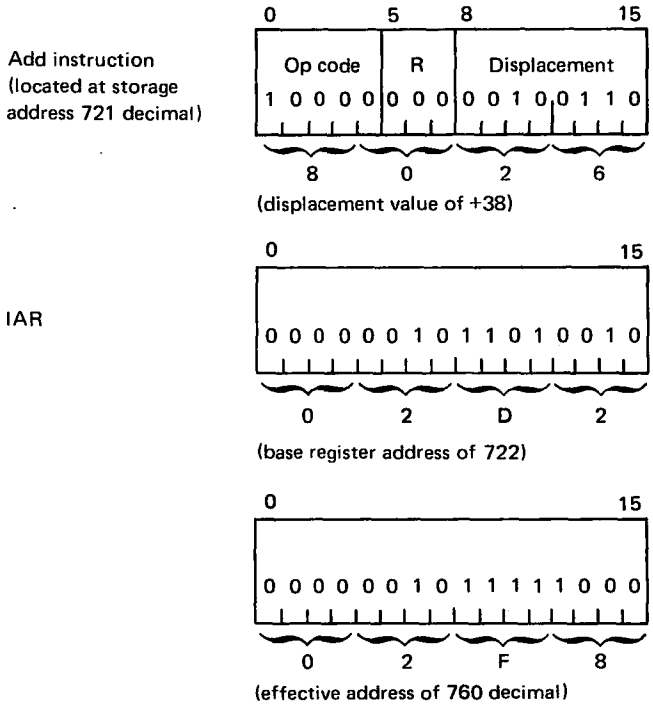
The address field of a long-format instruction is 16 bits long. Its contents are used in generating an effective address to locate the operand in four of the eight long-format instructions (load index long, branch conditional, branch and link long, branch and unmask long). The remaining four long-format instructions (execute I/O and the three storage protect instructions) use the address field to specify additional modifiers and/or the address of the I/O module and/or device.

EFFECTIVE ADDRESS GENERATION

Storage addresses are expressed internally in 16-bit binary numbers, ranging from 0 (16 bits set to 0) to a maximum value determined by the storage size of the System/7 processor module. Any attempt to address a location that is beyond the limits of the storage installed in the system causes a program check class interruption due to the invalid storage address. Most instructions refer to a storage address to locate the data that is to be used. This storage address is called the effective address (EA).

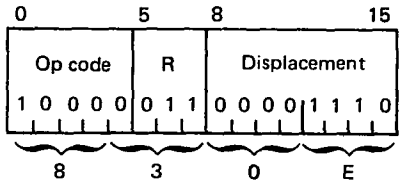
Address Arithmetic

The System/7 operates on a base-register-displacement principle for storage addressing. In a short-format instruction, the effective address is derived by adding algebraically the contents of the displacement field to the contents of the register indicated by the R field. The R field in the instruction specifies that the contents of the instruction address register (R=000) or one of the index registers (R=001 to 111) are added to the displacement value to compute the effective address. An example using the IAR (R=000) is shown below:



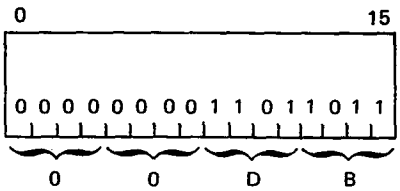
The data to be loaded into the accumulator is located by using the algebraic sum of the displacement field in the instruction and the contents of the instruction address register. In this example, the data at location 760 (decimal) would be loaded into the accumulator by this instruction. Another example, using index register 3 (R=011) is shown below:

Add instruction



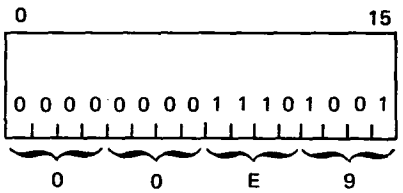
(displacement value of +14)

Index register 3



(base register address of 219)

The algebraic sum of the displacement field in the instruction and the contents of index register 3 is the effective address of the data that is to be used by the add instruction. In this case:



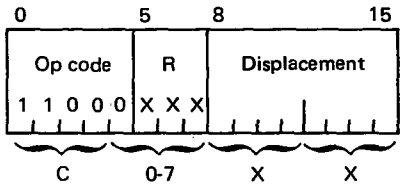
(effective address of 233)

The displacement can be either positive or negative; bit 8 of the instruction word is the sign bit for the displacement value. If this high-order bit of the displacement field is a 0, the displacement is positive with a maximum value of +127 (decimal). If the high-order bit of the displacement field is a 1, the displacement is negative with a maximum value of −128. The negative number is represented in twos-complement form.

The use of the address field varies for each long-format instruction. (Refer to the detailed descriptions of the 8 long-format instructions in this chapter: load index long, branch and link long, branch conditional, branch and unmask long, the 3 storage protect instructions, and execute I/O.)

Six instructions are provided to move data between storage and the index registers or accumulator in the processor module.

Load Accumulator (PL)

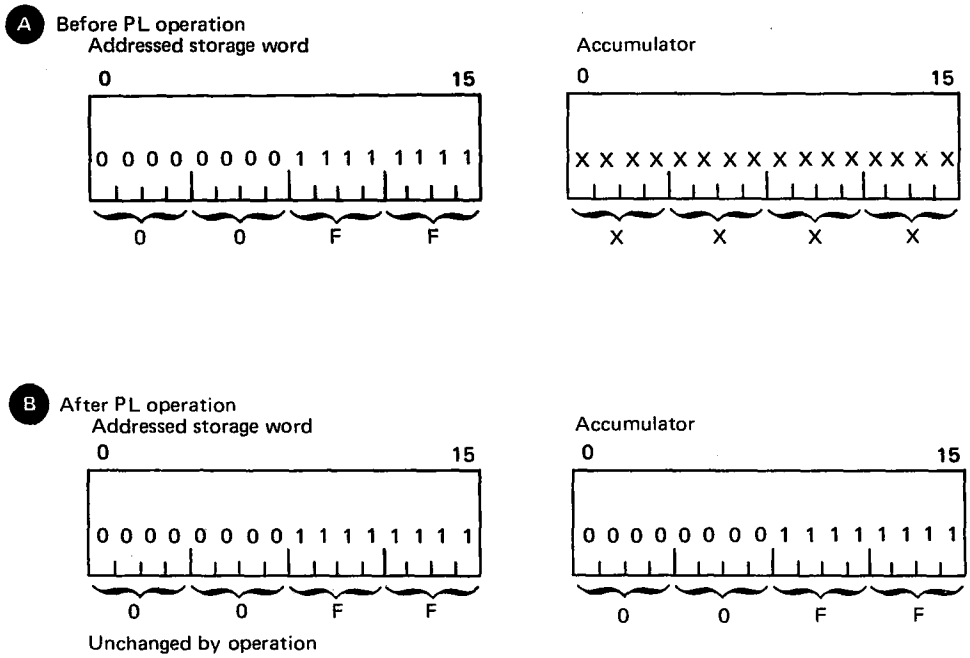


The contents of the storage location specified by the effective address replace the contents of the accumulator. The storage location contents are not changed.

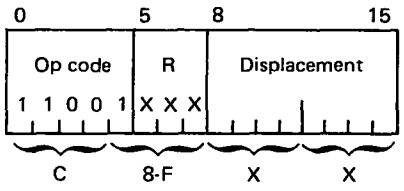
If R=000, the effective address is formed using the IAR as a base register; if R=001 to 111, an index register is used.

The carry and overflow indicators are not changed. The result indicators are changed to reflect the final contents of the accumulator.

Example:



Load and Zero (PLZ)

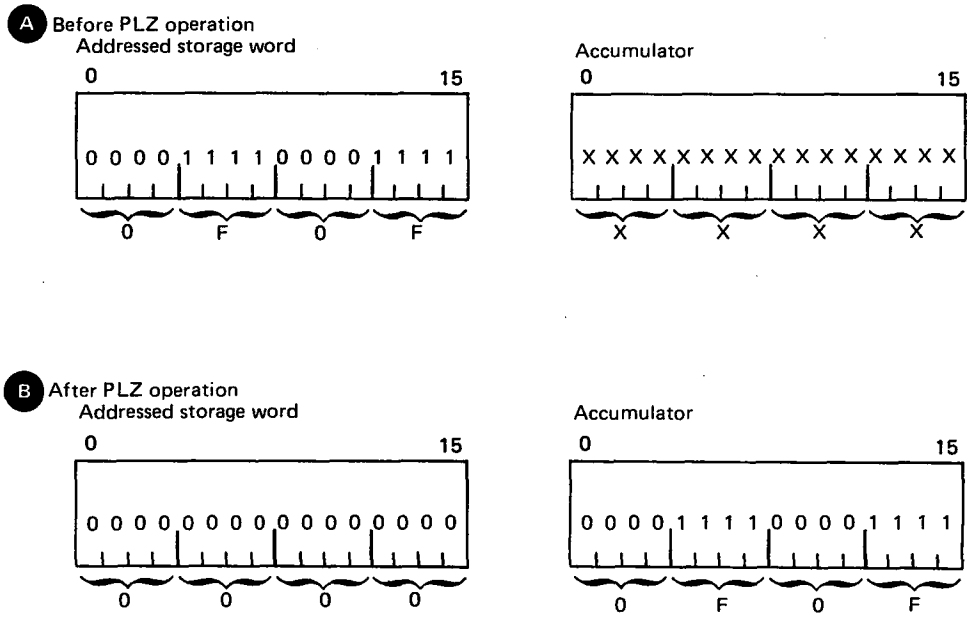


The contents of the storage location specified by the effective address replace the contents of the accumulator. The storage location contents are set to 0.

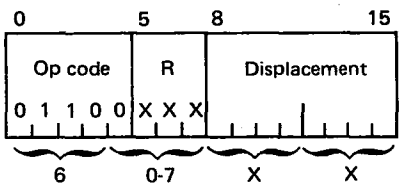
An R field of 000 means that the effective address is formed using the IAR as a base register; a nonzero R field (R=001 to 111) indicates that an index register is used to generate the EA.

The carry and overflow indicators are not changed. The result indicators are changed to reflect the final contents of the accumulator.

Example:



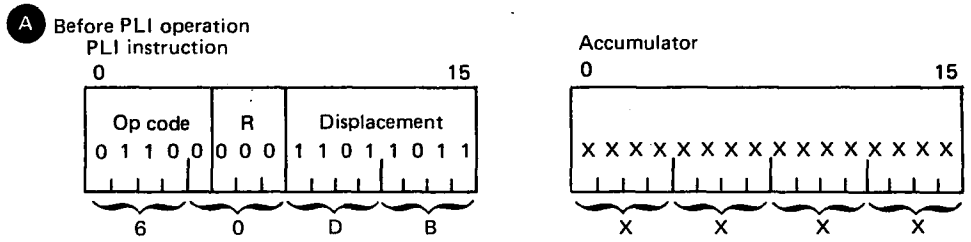
Load Immediate (PLI)

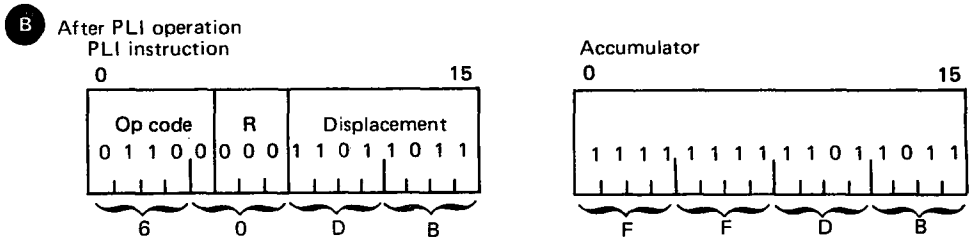


The contents of the displacement field of the instruction (bits 8 to 15) replace the contents of the register specified by the R field. An R field of 000 means that the displacement is loaded into the accumulator; a nonzero R field means that the displacement is loaded into one of the index registers. Before loading, the 8-bit displacement field is expanded to 16 bits by propagating the displacement sign-bit (bit 8) value through the high-order bits (that is, bits 0 to 7 take on the value of the sign bit). The instruction word is not changed.

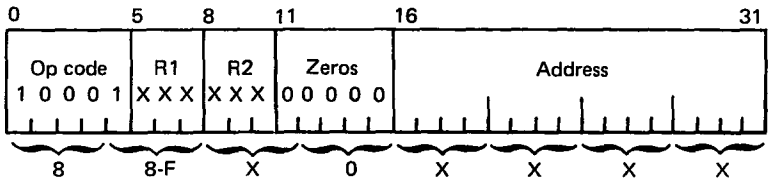
The carry and overflow indicators are not changed. The result indicators are changed to reflect the final contents of the accumulator or index register.

Example:





Load Index Long (PLXL)



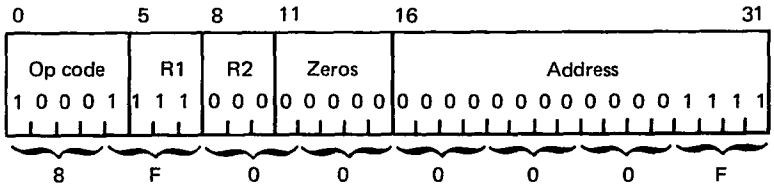
The contents of the storage location specified by the effective address replace the contents of the register specified by the R1 field. R1=000 specifies the accumulator; R1=001 to 111 specifies an index register. The storage contents are not changed.

If R2=000, the address field contains the effective address. If R2=001 to 111, the effective address is the algebraic sum of the contents of the index register (specified by R2) and the address field.

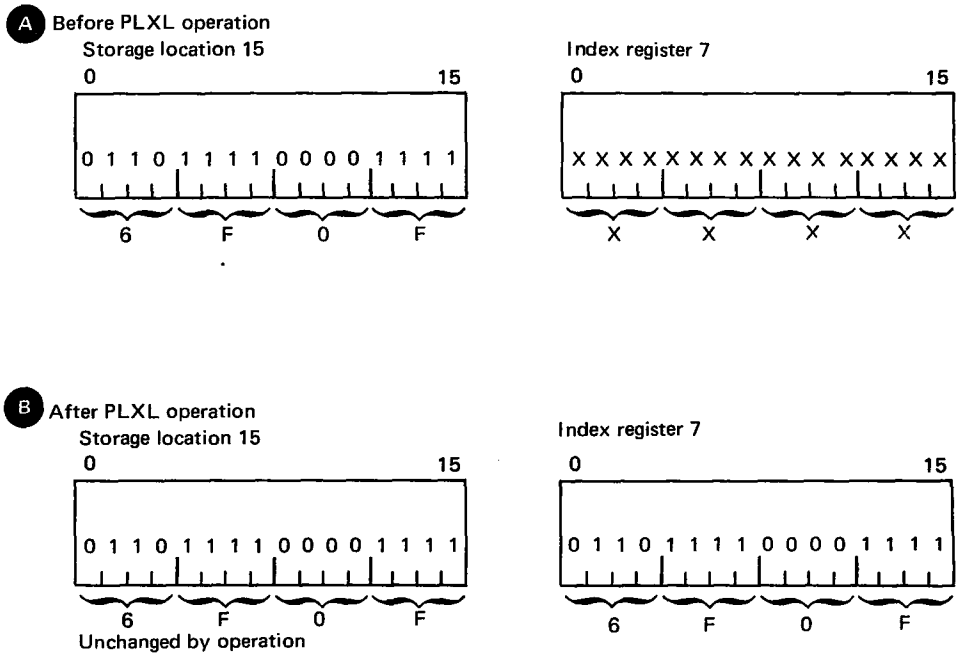
Bits 11 to 15 of the instruction must be 0's.

The carry and overflow indicators are not changed. The result indicators are changed to reflect the final contents of the accumulator or index register specified by R1.

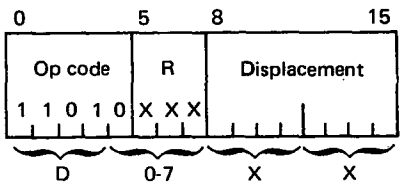
Example:



Since R2 = 000, the data is in storage location 15 (value of address field)



Store Accumulator (PST)

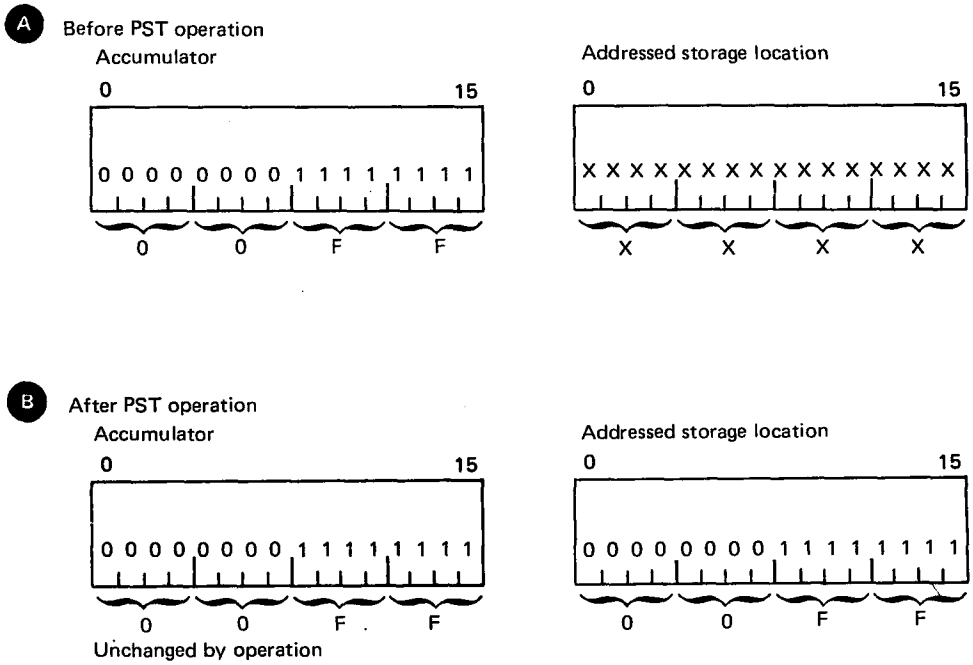


The contents of the accumulator replace the contents of the storage location specified by the effective address. The contents of the accumulator are not changed.

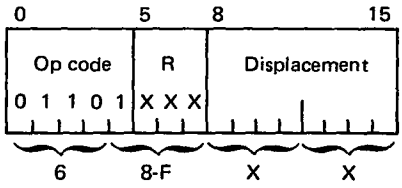
An R field of 000 means that the effective address is formed using the IAR as a base register; a nonzero R field (R=001 to 111) indicates that an index register is used to generate the EA.

The carry and overflow indicators are not changed. The result indicators are changed to reflect the final contents of the storage location.

Example:



Store Index (PSTX)



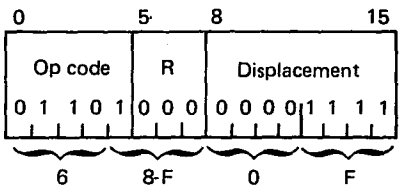
The contents of the index register (R) replace the contents of the storage location specified by the effective address. The index register contents are not changed. If R=000, the storage location contents are replaced by 0's.

The effective address of the storage location is always generated using the IAR as the base register.

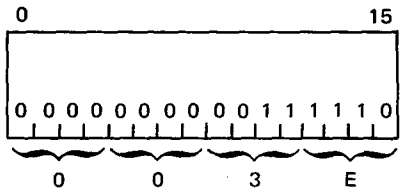
The carry and overflow indicators are not changed. The result indicators are changed to reflect the final contents of the storage location.

Example:

PSTX instruction (assumed to be at storage location 61)



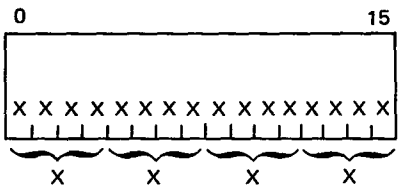
IAR during execution of PSTX instruction (contains address of next instruction at storage location 62)



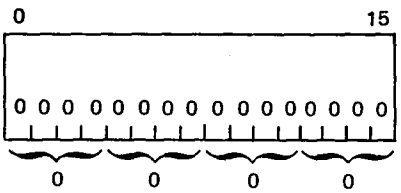
Effective address generation

IAR	0000 0000 0011 1110	Decimal 62
Displacement (expanded to 16 bits)	+0000 0000 0000 1111	Decimal 15
Effective address of storage location	0000 0000 0100 1101	Decimal 77

Storage location 77 before PSTX operation

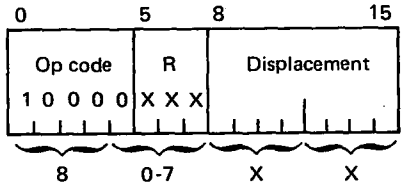


Storage location 77 after PSTX operation



Six instructions are provided to perform addition, subtraction, or complementing of data residing in storage and/or the index register or accumulator.

Add (PA)



The contents of the accumulator are added algebraically to the contents of the storage location specified by the effective address. The resulting sum replaces the contents of the accumulator; the contents of the addressed storage location are not changed.

If R=000, the effective address of the data in storage is formed by using the IAR as the base register. If the R field is nonzero (R=001 to 111), the effective address is formed using an index register as the base register.

The carry and result indicators are set to reflect the result in the accumulator.

The overflow indicator is set to reflect the result only if it has not been set by a previous operation. If the overflow indicator was previously set on, it will not be changed by the result of an add instruction. If an overflow occurs, the accumulator will contain the correct low-order 16 bits of the result and the carry indicator will be turned on if the result is negative.

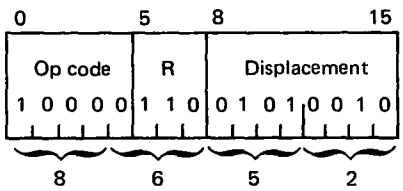
The result of the addition is either positive or negative, depending upon the magnitude of the values used and whether the signs of the two operands are the same, as shown below.

\+ added to a + = +
− added to a − = −
\+ added to a − = sign of the larger operand
− added to a + = sign of the larger operand

The value in the accumulator is positive if the leftmost bit is a 0; the value in the accumulator is negative if the leftmost bit is a 1. Negative numbers are in twos-complement form.

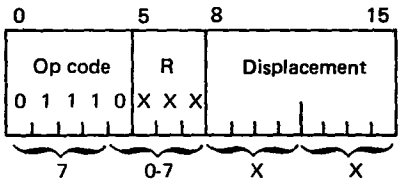
Example:

PA instruction
(R = 110 specifies using XR6
to generate the EA)



Contents of accumulator	0000 0000 1001 1101
Contents of storage location addressed by PA instruction	+ 0000 0010 0011 0101
Result in accumulator	0000 0010 1101 0010

Add Immediate (PAI)



The contents of the displacement field in the instruction are added algebraically to the contents of the register specified by R. An R field of 000 specifies the accumulator.

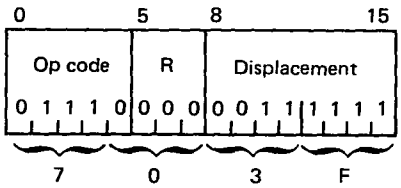
Before the addition takes place, the 8-bit displacement field is expanded to 16 bits by propagating the displacement sign-bit (bit 8) value through the high-order bits (that is, bits 0 to 7 take on the value of the sign bit). The resulting sum replaces the contents of the index register or accumulator. The instruction word is not changed.

The carry and result indicators are set to reflect the result in the index register or accumulator.

The overflow indicator is set to reflect the result only if it has not been set by a previous operation. If the overflow indicator was previously set on, it will not be changed by the result of an add immediate instruction. If an overflow occurs, the accumulator will contain the correct low-order 16 bits of the result and the carry indicator will be turned on if the result is negative.

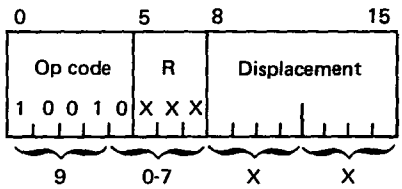
Example:

PAI instruction (R=000 specifies displacement is added to accumulator)



Contents of accumulator	0011 1111 0000 0000
Expanded displacement	+0000 0000 0011 1111
Result in accumulator	0011 1111 0011 1111

Subtract (PS)



The contents of the storage location specified by the effective address are subtracted algebraically from the contents of the accumulator. The resulting difference replaces the contents of the accumulator; the contents of the addressed storage location are not changed.

If R=000, the effective address of the data in storage is formed by using the IAR as the base register. If the R field is nonzero (R=001 to 111), the effective address is formed using an index register as the base register.

The carry indicator is set on if a borrow occurs out of the high-order bit in the accumulator.

The overflow indicator is set to reflect the result only if it has not been set by a previous operation. If the overflow indicator was previously set on, it will not be changed by the result of a subtract instruction. If an overflow occurs, the accumulator will contain the correct low-order 16 bits of the result and the carry indicator will be turned on if the result is negative.

The result indicators are set to reflect the result in the accumulator.

The sign of the result is dependent upon the signs and magnitudes of both operands. Possible combinations (where operand B is always numerically greater than operand A, regardless of signs) are:

From Operand in Accumulator	*Subtract Operand in Storage*		*Sign of Result in Accumulator*
+B	+A	=	+
+B	−A	=	+
−B	+A	=	−
−B	−A	=	−
+A	+B	=	−
+A	−B	=	+
−A	+B	=	−
−A	−B	=	+

The value in the accumulator is positive if the leftmost bit is a 0; the value in the accumulator is negative if the leftmost bit is a 1. Negative numbers are in twos-complement form.

Example 1:

```
Contents of accumulator                0 000 0000 0000 0011
Contents of storage location
addressed by PS instruction           -0 000 0000 0000 0010
Result in accumulator                  0 000 0000 0000 0001
```

(In decimal: 3 − 2 = 1)

Example 2:

```
Contents of accumulator                1 000 0000 0000 0011
Contents of storage location
addressed by PS instruction           -0 000 0000 0000 0010
Result in accumulator                  1 000 0000 0000 0001
```

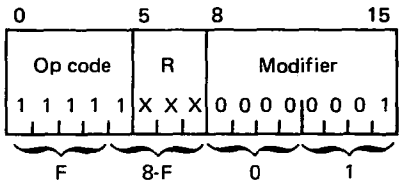
(In decimal: −32,765 − 2 = −32,767)

Example 3:

```
Contents of accumulator                1 000 0000 0000 0011
Contents of storage location
addressed by PS instruction           -1 000 0000 0000 0000
Result in accumulator                  0 000 0000 0000 0011
```

(In decimal: −32,765 − (−)32,768 = −32,765 + 32,768 = +3)

Add Register (PAR)



The contents of the register specified by R are added algebraically to the contents of the accumulator. The resulting sum replaces the contents of the accumulator. If R specifies an index register (R=001 to 111), the contents of the index register are not changed.

An R field of 000 specifies the accumulator. This means that the accumulator can be added to itself, giving a simple method of doubling the value of the accumulator contents.

The carry and result indicators are set to reflect the resulting sum in the accumulator.

The overflow indicator is set to reflect the result only if it has not been set by a previous operation. If the overflow indicator was previously set on, it will not be changed by the result of an add register instruction. If an overflow occurs, the accumulator will contain the correct low-order 16 bits of the result and the carry indicator will be turned on if the result is negative.

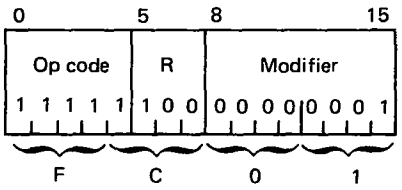
The result of the addition is either positive or negative, depending upon the magnitude of the values used and whether the signs of the two operands are the same, as shown below.

\+ added to a + = +
− added to a − = −
\+ added to a − = sign of the larger operand
− added to a + = sign of the larger operand

The value in the accumulator is positive if the leftmost bit is a 0; the value in the accumulator is negative if the leftmost bit is a 1. Negative numbers are in twos-complement form.

Example:

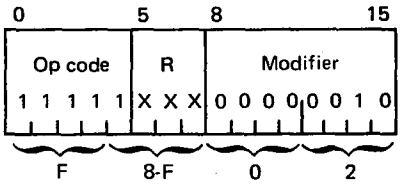
PAR instruction (R=100 specifies index register 4 is added to accumulator)



Contents of accumulator	0011 0011 1101 1111
Contents of index register 4	+0001 1110 0001 1111
Result in accumulator	0101 0001 1111 1110

(in decimal: 13,279 + 7,711 = 20,990)

Subtract Register (PSR)



The contents of the register specified by R are subtracted algebraically from the contents of the accumulator. The resulting difference replaces the contents of the accumulator. If R specifies an index register (R=001 to 111), the contents of the index register are not changed.

An R field of 000 specifies the accumulator. This means that the accumulator can be subtracted from itself, giving a simple method of setting the accumulator contents to a 0 value.

The carry and result indicators are set to reflect the resulting difference in the accumulator. The carry indicator is set on by a borrow from the high-order bit in the accumulator.

The overflow indicator is set to reflect the result only if it has not been set by a previous operation. If the overflow indicator was previously set on, it will not be changed by the result of a subtract register instruction. If an overflow occurs, the accumulator will contain the correct low-order 16 bits of the result and the carry indicator will be turned on if the result is negative.

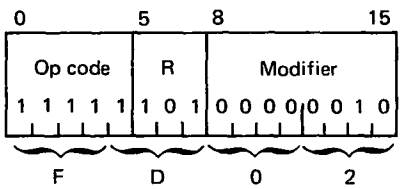
The sign of the result is dependent upon the signs and magnitudes of both operands. Possible combinations (where operand B is always numerically greater than operand A, regardless of signs) are:

From Operand in Accumulator	*Subtract Operand in Storage*		*Sign of Result in Accumulator*
+B	+A	=	+
+B	−A	=	+
−B	+A	=	−
−B	−A	=	−
+A	+B	=	−
+A	−B	=	+
−A	+B	=	−
−A	−B	=	+

The value in the accumulator is positive if the leftmost bit is a 0; the value in the accumulator is negative if the leftmost bit is a 1. Negative numbers are in twos-complement form.

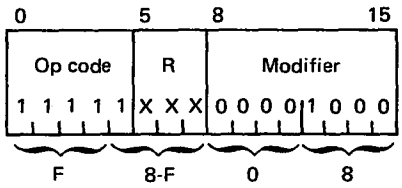
Example:

PSR instruction
(R = 101 specifies index register
5 is subtracted from accumulator)



Contents of accumulator — 0000 0000 1110 1111
Contents of index register 5 — −0000 0000 0011 0101
Result in accumulator — 0000 0000 1011 1010
(in decimal: 239−53 = 186)

Complement Register (PCR)



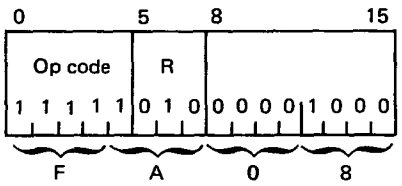
The twos-complement of the register specified by R replaces the contents of the same register. An R field of 000 specifies the accumulator.

The carry indicator is not changed. The overflow indicator is set on if the number to be complemented is the maximum negative number that can be represented (a 1 followed by fifteen 0's). The result indicators are changed to reflect the final contents of the accumulator or index register.

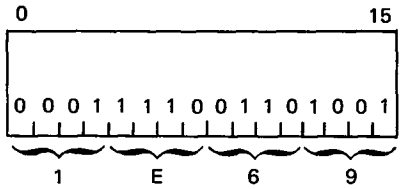
The twos-complement of a number is obtained by inverting each bit of the number and then adding a binary 1 to the result.

Example:

PCR instruction (R=010 specifies index register 2 is to be complemented)



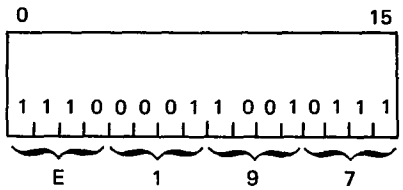
(A) Before PCR operation
Index register 2



Complement procedure:

Invert bits	1110 0001 1001 0110
Add +1	+0000 0000 0000 0001
Twos - complement	1110 0001 1001 0111

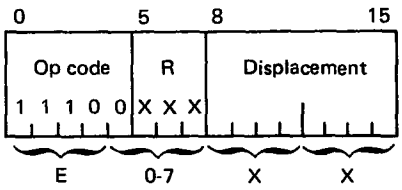
(B) After PCR operation
Index register 2



LOGICAL INSTRUCTIONS

A logical instruction is one that is defined in symbolic logic, such as AND, OR, and exclusive OR. Six instructions are provided to perform logical operations with data in storage and/or the index registers or the accumulator. Two words of data are used by these instructions in order to produce a one-word logical result.

AND (PN)



The contents of the accumulator are ANDed, bit by bit, with the contents of the storage location specified by the effective address. The result replaces the contents of the accumulator. The contents of the addressed storage word are not changed.

An R field of 000 means that the effective address of the storage data is generated using the IAR as the base register.

The carry and overflow indicators are not changed. The result indicators are changed to reflect the final contents of the accumulator.

ANDing occurs only between corresponding bits in the accumulator and the storage word: bit 0 is ANDed only with bit 0, bit 1 only with bit 1, and so on. The four possible ANDing combinations are:

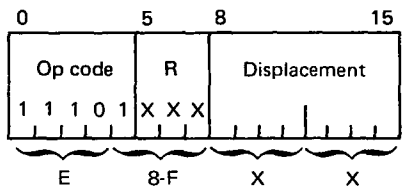
Bit Value from Storage Word	*Bit Value from Accumulator*	*Result in Accumulator*
0	0	0
0	1	0
1	0	0
1	1	1

Thus, a bit value of 1 will result in the accumulator only if a corresponding bit position is set to a value of 1 in both the accumulator *and* the storage word.

Example:

Word in accumulator	0101 0000 1111 1010
Word from storage	1010 1111 1010 1111
Result in accumulator	0000 0000 1010 1010

OR (PO)



The contents of the accumulator are ORed, bit by bit, with the contents of the storage location specified by the effective address. The result replaces the contents of the accumulator. The contents of the addressed storage word are not changed.

An R field of 000 means that the effective address of the storage data is generated using the IAR as the base register.

The carry and overflow indicators are not changed. The result indicators are changed to reflect the final contents of the accumulator.

ORing occurs only between corresponding bits in the accumulator and the storage word: bit 0 is ORed only with bit 0, bit 1 only with bit 1, and so on. The four possible ORing combinations are:

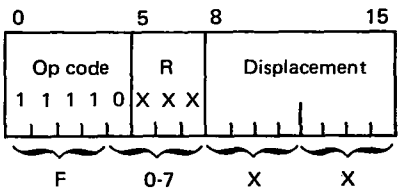
Bit Value from Storage Word	*Bit Value from Accumulator*	*Result in Accumulator*
0	0	0
0	1	1
1	0	1
1	1	1

Thus, a bit value of 1 will result in the accumulator if a corresponding bit position is set to a value of 1 in either the accumulator *or* the storage word. If the bit positions in both the accumulator and the storage word are set to a value of 1, the OR condition is still satisfied, so the corresponding result bit is set to a 1 in the accumulator.

Example:

Word in accumulator	0011 0101 1111 1010
Word from storage	0101 0001 1010 0000
Result in accumulator	0111 0101 1111 1010

Exclusive OR (PX)



The contents of the accumulator are exclusive ORed, bit by bit, with the contents of the storage location specified by the effective address. The result replaces the contents of the accumulator. The contents of the addressed storage word are not changed.

An R field of 000 means that the effective address of the storage data is generated using the IAR as the base register.

The carry and overflow indicators are not changed. The result indicators are changed to reflect the final contents of the accumulator.

Exclusive ORing occurs only between corresponding bits in the accumulator and the storage word: bit 0 is exclusive ORed only with bit 0, bit 1 only with bit 1, and so on. The four possible exclusive-ORing combinations are:

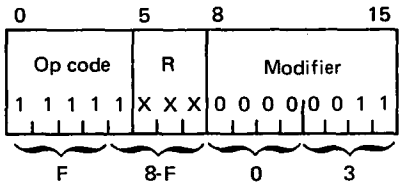
Bit Value from Storage Word	*Bit Value from Accumulator*	*Result in Accumulator*
0	0	0
0	1	1
1	0	1
1	1	0

Thus, a bit value of 1 will result in the accumulator only if a corresponding bit position is set to a value of 1 in the accumulator *or* in the storage word, but not both.

Example:

Word in accumulator	0110 1100 0000 1111
Word from storage	1100 0011 0000 1111
Result in accumulator	1010 1111 0000 0000

AND Register (PNR)



The contents of the accumulator are ANDed, bit by bit, with the contents of the register (R). The result replaces the contents of the accumulator. The contents of the addressed register are not changed.

If R=000, specifying the accumulator, the instruction is, in effect, a no-operation because the final contents of the accumulator are unchanged from the original contents. Any data ANDed with itself has a final result equal to the original data.

The carry and overflow indicators are not changed. The result indicators are changed to reflect the final contents of the accumulator.

ANDing occurs only between corresponding bits in the accumulator and in the index register: bit 0 is ANDed only with bit 0, bit 1 only with bit 1, and so on. The four possible ANDing combinations are:

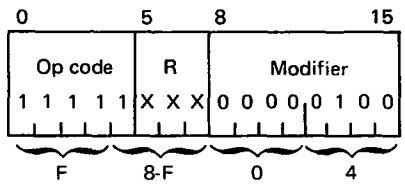
Bit Value from Index Register	*Bit Value from Accumulator*	*Result in Accumulator*
0	0	0
0	1	0
1	0	0
1	1	1

Thus, a bit value of 1 will result in the accumulator only if a corresponding bit position is set to a value of 1 in both the accumulator *and* the index register.

Example:

Word in accumulator	0101 0000 1111 1010
Word in index register	1010 1111 1010 1111
Result in accumulator	0000 0000 1010 1010

OR Register (POR)



The contents of the accumulator are ORed, bit by bit, with the contents of the register (R). The result replaces the contents of the accumulator. The contents of the addressed register are not changed.

If R=000, specifying the accumulator, this instruction is, in effect, a no-operation because the final contents of the accumulator are unchanged from the original contents. Any data ORed with itself has a final result equal to the original data.

The carry and overflow indicators are not changed. The result indicators are changed to reflect the final contents of the accumulator.

ORing occurs only between corresponding bits in the accumulator and the R register: bit 0 is ORed only with bit 0, bit 1 only with bit 1, and so on. The four possible ORing combinations are:

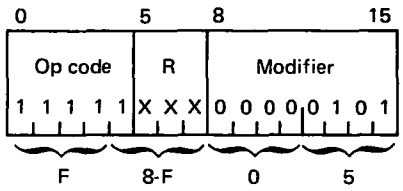
Bit Value from Index Register	*Bit Value from Accumulator*	*Result in Accumulator*
0	0	0
0	1	1
1	0	1
1	1	1

Thus, a bit value of 1 will result in the accumulator if a corresponding bit position is set to a value of 1 in either the accumulator *or* the register (R). If the bit positions in both the accumulator and the R register are set to a value of 1, the OR condition is still satisfied, so the corresponding result bit is set to a 1 in the accumulator.

Example:

Word in accumulator	0011 0101 1111 1010
Word in index register	0101 0001 1010 0000
Result in accumulator	0111 0101 1111 1010

Exclusive OR Register (PXR)



The contents of the accumulator are exclusive ORed, bit by bit, with the contents of the register (R). The result replaces the contents of the accumulator. When R specifies an index register, the contents of the addressed index register are not changed.

An R field of 000, which specifies the accumulator, is a simple method of clearing the accumulator. Any data that is exclusive ORed with itself results in setting all the bits to 0's.

The carry and overflow indicators are not changed. The result indicators are changed to reflect the final contents of the accumulator.

Exclusive ORing occurs only between corresponding bits in the accumulator and the specified register: bit 0 is exclusive ORed only with bit 0, bit 1 only with bit 1, and so on. The four possible exclusive-ORing combinations are:

Bit Value from Index Register	*Bit Value from Accumulator*	*Result in Accumulator*
0	0	0
0	1	1
1	0	1
1	1	0

Thus, a bit value of 1 will result in the accumulator only if a corresponding bit position is set to a value of 1 in the accumulator *or* in the register (R), but not both.

Example:

Word in accumulator	0110 1100 0000 1111
Word in index register	1100 0011 0000 1111
Result in accumulator	1010 1111 0000 0000

SHIFT INSTRUCTIONS

Shift instructions shift data, bit by bit, to the right or to the left. The shift operation can be performed on data residing in either an index register or the accumulator. All shift instructions have the same operation code in bits 0 to 4. A particular shift operation is specified by the modifier bits (bits 8 to 10) of the shift instruction.

The manner of shifting depends on the particular shift instruction. In shift-left operations, bits that are shifted into the low-order vacated position can be:

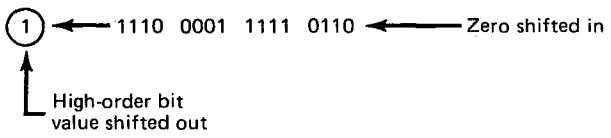
1. Zeros—a logical left shift (PSLL instruction).
2. Bits that are shifted out of the high-order position (bit 0) of the shifted register—a circular left shift (PSLC instruction).

For example:

Original operand in accumulator

1111 0000 1111 1011

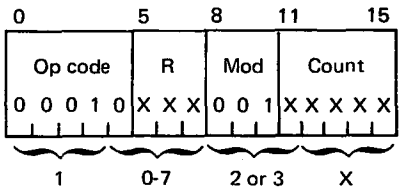
After a logical shift left of one position



In shift-right operations, bits that are shifted into the vacated high-order position can be:

1. Zeros—a logical right shift (PSRL instruction).
2. The original value of the sign bit (bit 0 of the shifted register)—an arithmetic right shift (PSRA instruction).

Shift Left Logical (PSLL)



All 16 bits in the index register (R), or the accumulator if R=000, are shifted left by the number specified in the count field.

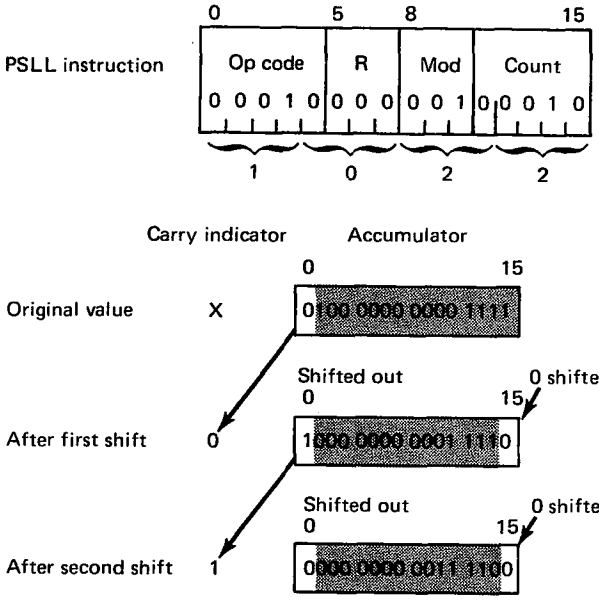
All bits shifted out of the high-order position (bit 0) are lost. The last bit shifted out of position 0 (shift count=0) sets the carry indicator according to the value of the bit shifted out. Vacated low-order bit positions are set to 0's.

The shift count field can specify any decimal value from 0 to 16. Shift counts greater than 16 are invalid and cause a program check interruption. A shift count of 0 is valid and serves a useful purpose. Although no shifting takes place, this is a simple method of setting the result indicators to reflect the current contents of the R register. A shift count of 16 puts 0's in all bits of the R register.

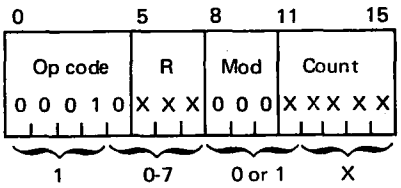
The overflow indicator is not changed. The carry indicator is set on or off to reflect the last bit shifted out of bit 0. The result indicators are changed to reflect the final contents of the R register. For a shift count greater than 0, the result-even indicator is always set on, because vacated bits are automatically set to 0's.

Example:

Assume a left shift of two bits in the accumulator



Shift Left Circular (PSLC)



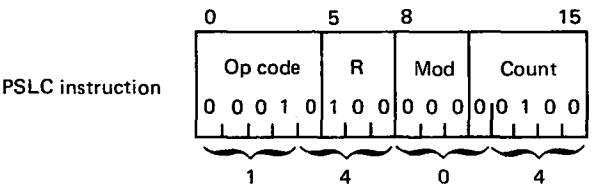
All 16 bits in the index register (R), or the accumulator if R=000, are rotated left by the number of bits specified in the count field. This means that bits shifted out of the high-order position (bit 0) reenter at the low-order position (bit 15). Therefore, no bits are lost; they are just rearranged.

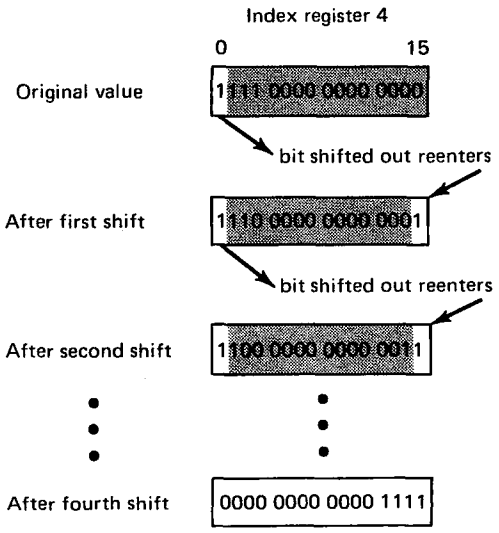
The shift count field can specify any decimal value from 0 to 16. Shift counts greater than 16 are invalid and cause a program check interruption. A shift count of 0 is valid and serves a useful purpose. Although no rotating takes place, this is a simple method of setting the result indicators to reflect the current contents of the R register. A shift count of 16 has the same effect because the bits are rotated until they return to their original positions.

The carry and overflow indicators are not changed. The result indicators are changed to reflect the final contents of the R register.

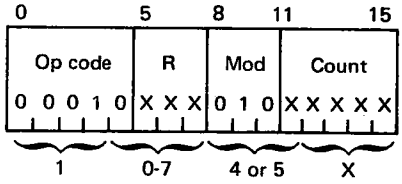
Example:

Assume a left circular shift of four bits in index register 4





Shift Right Logical (PSRL)



All 16 bits in the index register (R), or the accumulator if R=000, are shifted right by the number of bits specified in the count field.

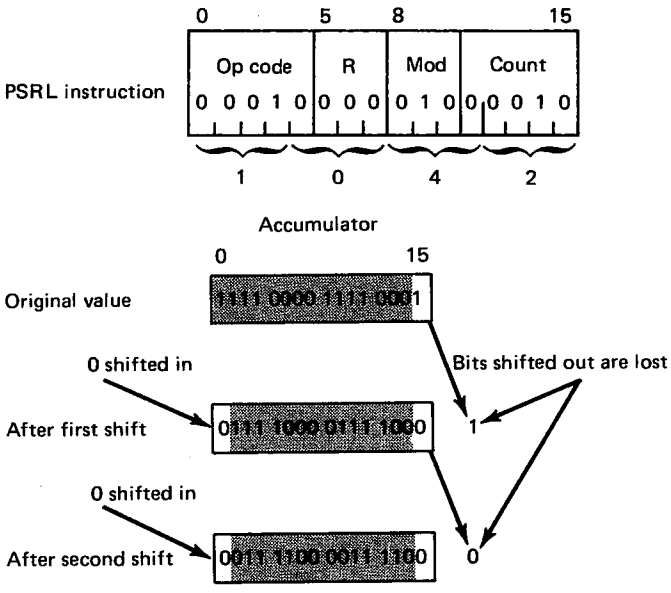
Vacated high-order bits are set to 0's. All bits shifted out of the low-order position (bit 15) are lost.

The shift count field can specify any decimal value from 0 to 16. Shift counts greater than 16 are invalid and cause a program check interruption. A shift count of 16 sets the entire contents of the R register to a 0 value. A shift count of 0 is valid and serves a useful purpose. Although no shifting takes place, this is a simple method of setting the result indicators to reflect the current contents of the R register.

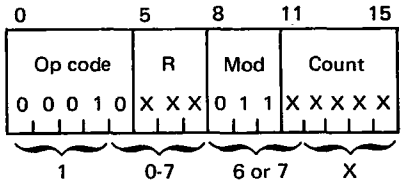
The carry and overflow indicators are not changed. The result indicators are changed to reflect the final contents of the R register. For a shift count greater than 0, the result-positive indicator is always set on.

Example:

Assume a right shift of two bits in the accumulator



Shift Right Arithmetic (PSRA)



All 16 bits in the index register (R), or the accumulator if R=000, are shifted right by the number of bits specified in the count field.

Vacated high-order bits are set to the value of the sign bit (bit 0). All bits shifted out of the low-order bit (bit 15) are lost.

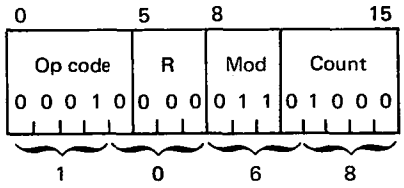
The shift count field can specify any decimal value from 0 to 16. Shift counts greater than 16 are invalid and cause a program check interruption. A shift count of 15 or 16 sets the entire contents of the R register to the value of the sign bit. A shift count of 0 is valid and serves a useful purpose. Although no shifting takes place, this is a simple method of setting the result indicators to reflect the current contents of the R register.

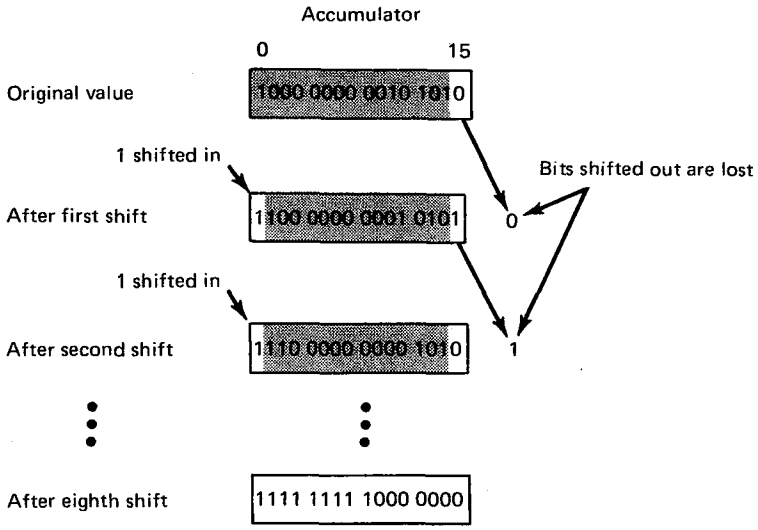
The carry and overflow indicators are not changed. The result indicators are changed to reflect the final contents of the R register.

Example:

Assume a right arithmetic shift of eight bits in the accumulator

PSRA instruction

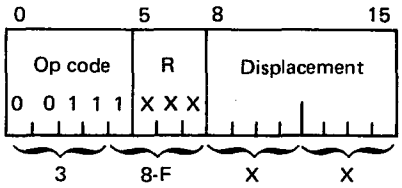




BRANCH AND SKIP INSTRUCTIONS

Branch instructions are used to depart (branch) from a sequential series of instructions and, in some cases, save the storage address from which the departure was made. Skip instructions are used to skip over the storage word immediately following the skip instruction. One skip instruction and one branch instruction test for the presence or absence of one or more conditions to determine if the branch or skip actually should be taken.

Branch (PB)



The branch instruction (PB) terminates the execution of a sequential series of instructions and branches to another location in storage. The PB instruction is unconditional; the branch is always taken because it is not based on the result of a condition or test.

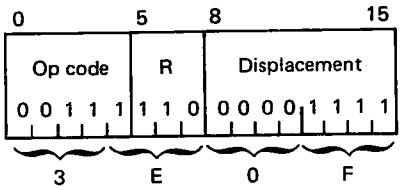
Location of the next instruction to be executed (the location branched to) is determined by the effective address as computed from the register and displacement fields in the PB instruction. The contents of the displacement field are added algebraically to the contents of the specified register (R) to form this effective address. If R=000, the IAR is used in this calculation. Since the IAR always contains the address of the next instruction to be executed, the PB instruction replaces the contents of the IAR with the calculated effective address. A branch (PB) instruction with R=000 and a displacement of 0, therefore, performs no operation at all, but proceeds with the same instruction sequence.

The PB instruction does not change the carry, overflow, and result indicators.

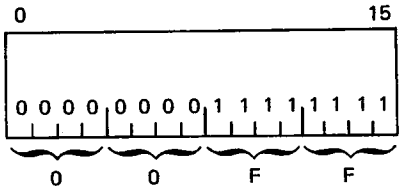
Example:

PB instruction

(R=110 specifies index register 6 is to be used for calculating the branch-to location)



Index register 6

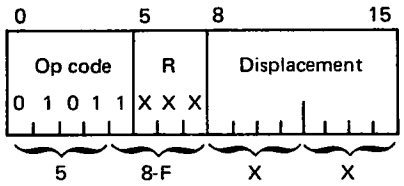


Effective address generation:

Contents of index register 6	0000 0000 1111 1111
Expanded displacement	+0000 0000 0000 1111
Effective address	0000 0001 0000 1110

Thus, the next instruction to be executed is at storage location 270 (decimal).

Branch and Link (PBAL)



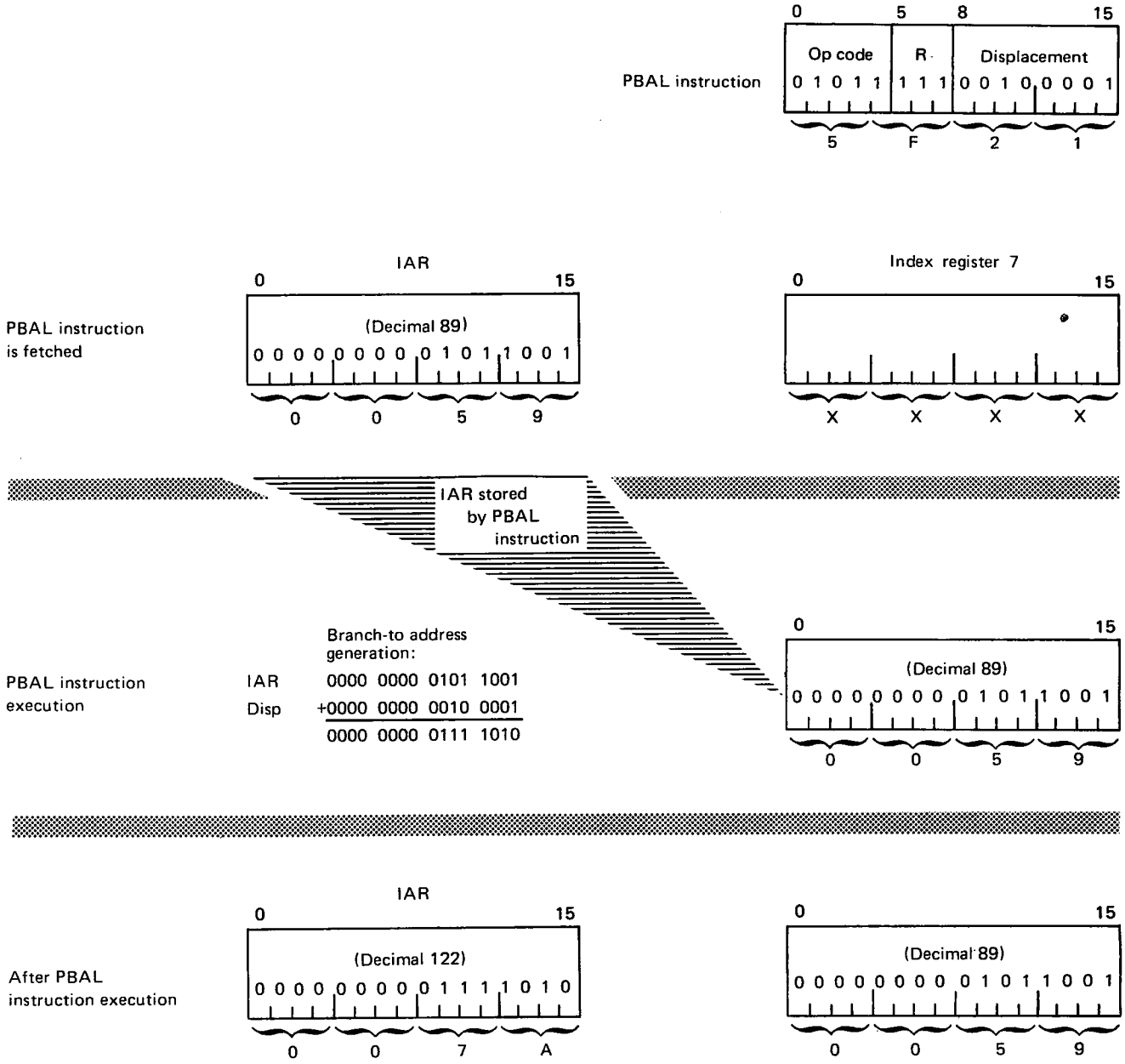
The branch and link instruction (PBAL) branches to another storage location and saves the contents of the IAR for return to the original sequence. The PBAL instruction is unconditional; the branch is always taken because it is not based upon the result of a condition or test.

When the PBAL instruction is fetched from storage, the IAR contains the address of the storage location that immediately follows the PBAL instruction. Execution of the PBAL instruction causes the contents of the IAR to be stored (and thus saved for future use in returning to the original sequence) in the index register specified by the R field. If R=000, the accumulator is used for this purpose.

The location of the next instruction to be executed (the location branched to) is determined by the effective address as computed from the IAR and displacement fields in the PBAL instruction. The contents of the displacement field are added algebraically to the contents of the IAR to form this effective address. Thus, the IAR now contains the address of the storage location being branched to. The PBAL instruction does not change the carry, overflow, or result indicators.

The branch and link instruction (PBAL) permits branching only to storage locations that are +127 or −128 positions away from the address contained in the IAR when the PBAL instruction is fetched. This limit exists because the displacement field (bits 8 to 15) of the instruction can contain only numbers in the range of +127 through −128.

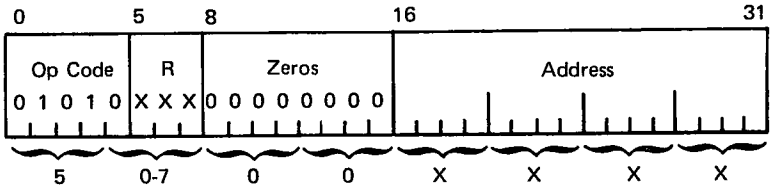
Example: Assume that the PBAL instruction is at storage location 88.



The IAR contains the effective address of the next instruction to be executed; that is, the instruction branched to, which is at location 122. The branch to location 122 is taken and execution begins. The PBAL instruction does not change the carry, overflow, or result indicators.

When it is desired to return to the original sequence (at location 89), the contents of index register 7 must be moved to the IAR. A branch (PB) instruction, with 0 displacement and R=111 (specifying XR7), can be used to accomplish this.

Branch and Link Long (PBALL)



The branch and link long instruction (PBALL) branches to another storage location and saves the contents of the IAR for return to the original sequence. The PBALL instruction is unconditional; the branch is always taken because it is not based upon the result of a condition or test.

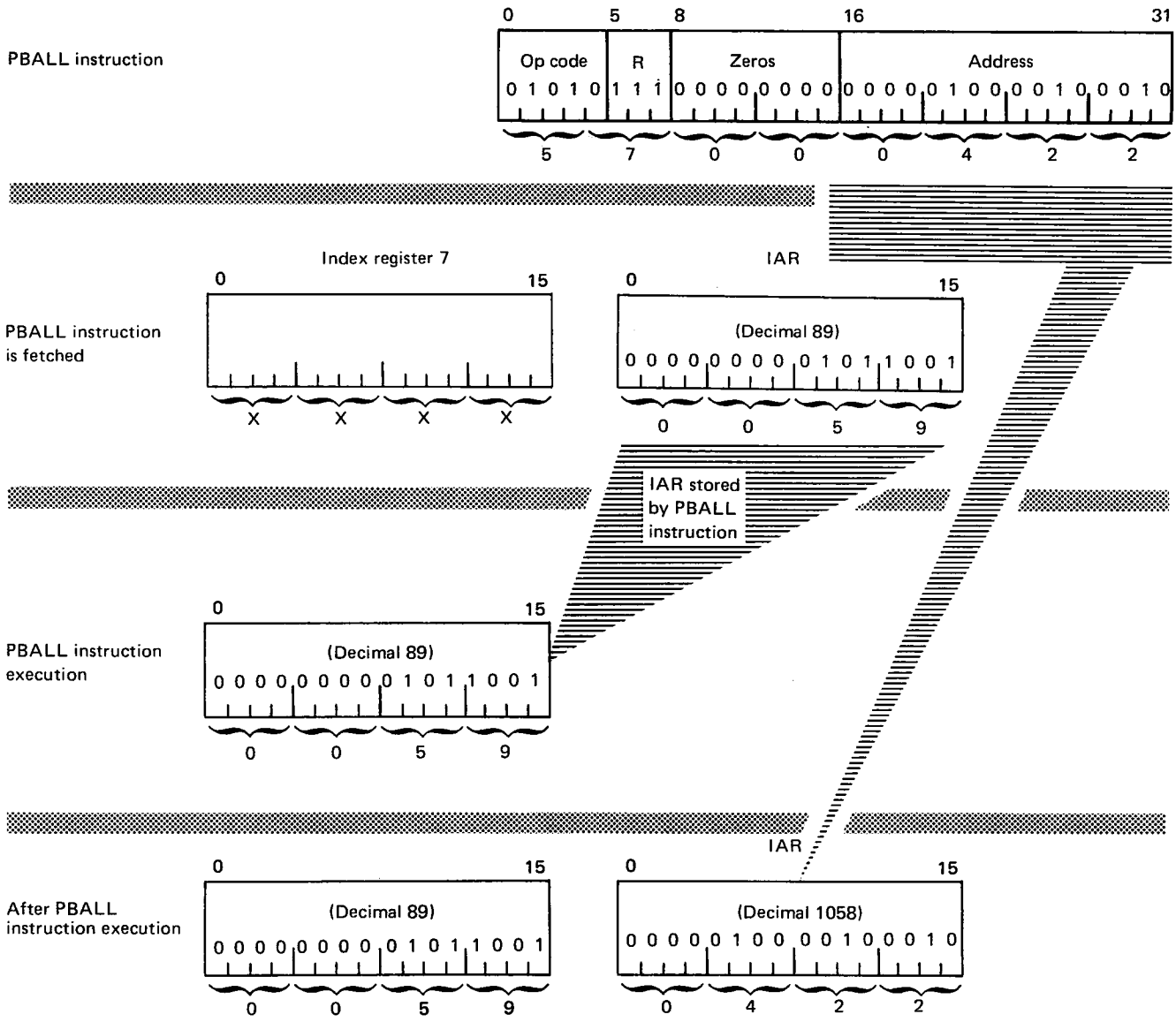
When the second word of the PBALL instruction is fetched from storage, the IAR contains the address of the storage location immediately following the PBALL instruction. Execution of the PBALL instruction causes the contents of the IAR to be stored (and thus saved for future use in returning to the original sequence) in the index register specified by the R field. If R=000, the accumulator is used for this purpose.

The location of the next instruction to be executed (the location branched to) is contained in the address field of the PBALL instruction. This address is loaded into the IAR and the branch is made to that address. The branch and link long instruction (PBALL) permits branching to *any* location in main storage, whereas the branch and link instruction (PBAL) limits branching to locations that are +127 or −128 storage positions away from the address contained in the IAR when the PBAL instruction is executed.

Bits 8 to 15 of the instruction must always be set to 0's.

The branch and link long instruction does not change the carry, overflow, and result indicators.

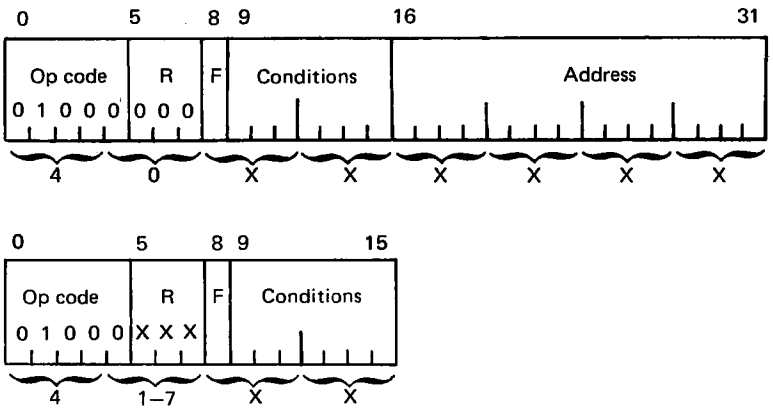
Example: Assume that the PBALL instruction is at storage locations 87 and 88.



The IAR contains the effective address of the next instruction to be executed; that is, the instruction branched to, which is at location 1058. The branch to location 1058 is taken and execution begins.

When it is desired to return to the original sequence (at location 89), the contents of index register 7 must be moved to the IAR. A branch (PB) instruction, with 0 displacement and R=111 (specifying XR7), can be used to accomplish this.

Branch Conditional (PBC)



The branch conditional instruction (PBC) branches to another storage location only if certain conditions are not present. These conditions are specified within the PBC instruction. If any one of the specified conditions is present, the PBC instruction does not branch, but continues program execution with the instruction that immediately follows the PBC instruction.

There are two ways of determining the effective address of the storage location that is to be branched to. When R=000, the effective address is in the address field of a long-format PBC instruction. When R does not equal 000, the effective address is contained in the specified index register (R) and the short-format PBC instruction is used. In either case, if test results cause the branch to be taken, the effective address is loaded into the IAR, and the instruction sequence begins at the effective address.

The programmer determines what tests are conducted to decide whether a branch will be taken. This is accomplished by setting bits on or off in the condition field (bits 8 to 15) of the instruction.

The bits and their meanings are shown below:

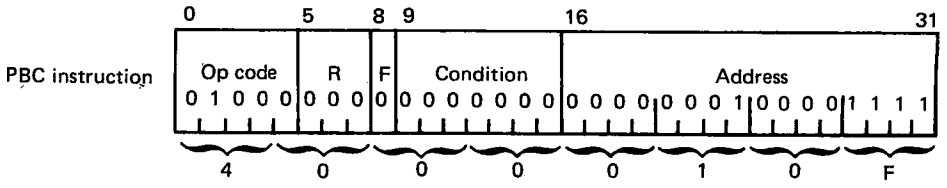
Bit Set On	*Condition Tested*
9	Carry and overflow indicators both off
10	Zero-result indicator on
11	Negative-result indicator on
12	Positive-result indicator on
13	Even-result indicator on
14	Carry indicator off
15	Overflow indicator off

If no conditions are tested (all condition field bits are set to 0's), the PBC functions as an unconditional branch. Any combination of testing can be requested.

If none of the specified test conditions are present, the branch is taken. If any are present, the branch does not occur and program execution proceeds with the instruction that immediately follows the PBC instruction.

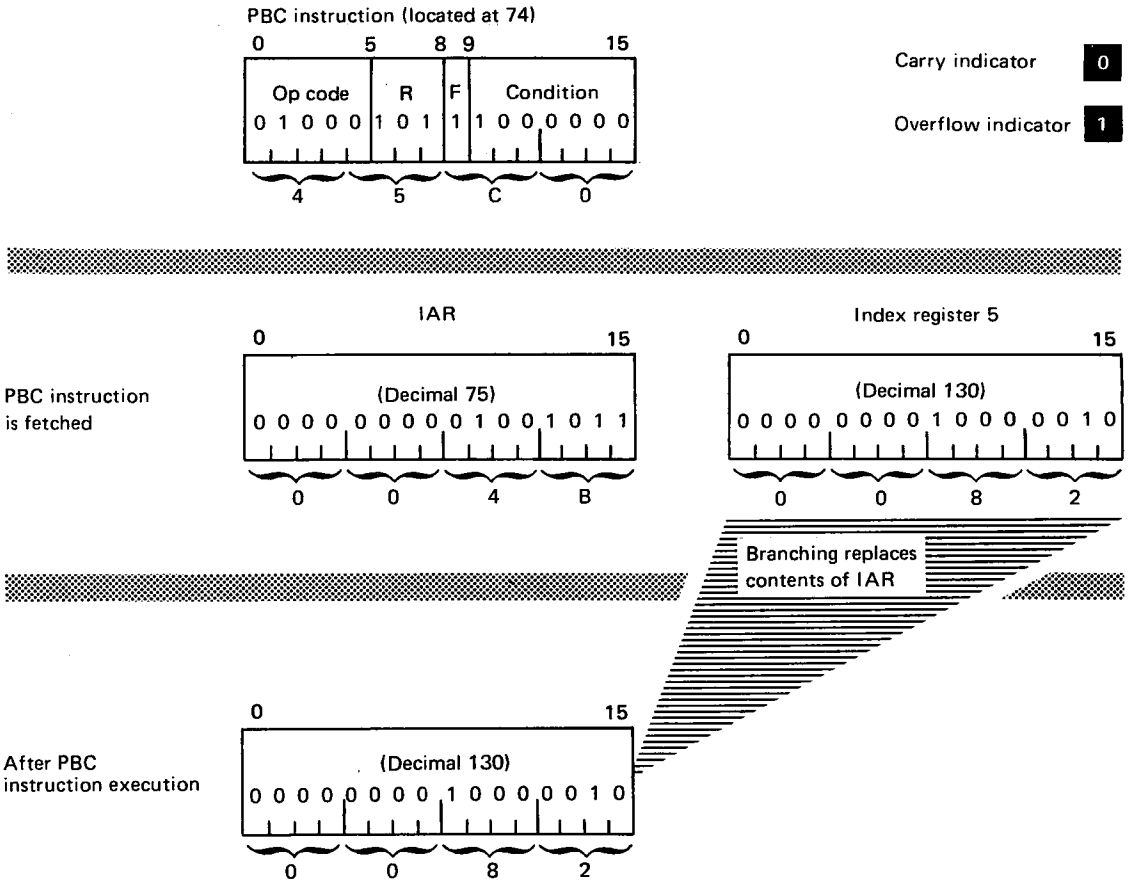
The branch conditional instruction does not change the carry and result indicators. Testing the overflow indicator with bit 15 always sets the overflow indicator off unless condition field bit 8 (overflow-save flag) of the instruction is set on. Testing the overflow indicator with bit 9 does not change the overflow indicator; the overflow-save flag has no effect.

Example 1:



No testing is requested since all condition field bits are 0's. The effective address is in the address field because R=000. Therefore, this instruction is an unconditional branch to storage location 271.

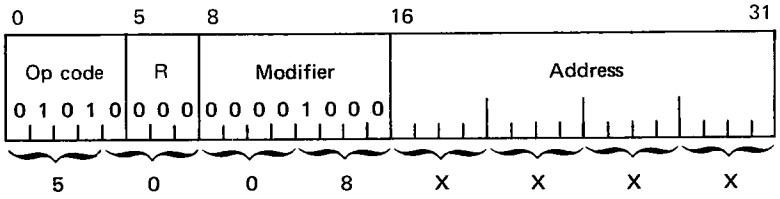
Example 2:



The only test requested (bit 9 set on) is that both the carry and overflow indicators are off. This is not true because the overflow indicator is on. Therefore, this instruction results in a branch to location 130, the effective address contained in index register 5.

Example 3: In example 2, if the overflow indicator is off, the test requested is true. In that case, no branch occurs. Program execution proceeds with the next sequential instruction following the short-format branch conditional instruction.

Branch and Unmask Long (BUL) (5010 Processor Module Model E only)



The branch and unmask long instruction branches to another storage location specified by the contents of the address field. The BUL instruction is unconditional; the branch is always taken because it is not based upon the results of a condition or test.

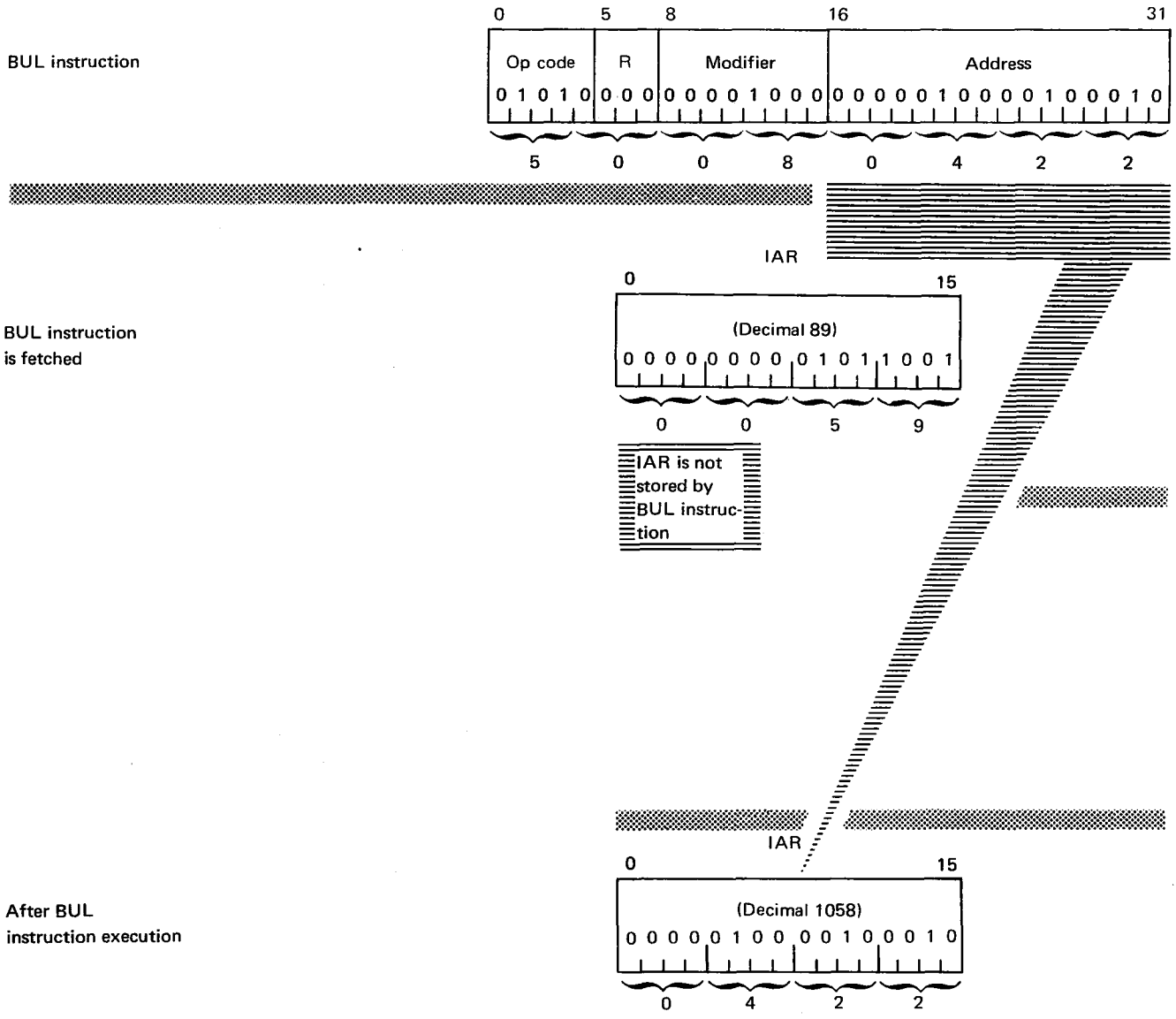
The contents of the IAR are replaced by the contents of the address field and become the location branched to. The contents of the IAR are not saved.

The summary mask is turned off, thus returning interrupt servicing to levels enabled in the interruption mask register. Interruption of the program may occur before, or immediately after, the next sequential instruction.

The R field is unused and must be set to 0's. The branch and unmask long instruction does not change the carry, overflow, and result indicators.

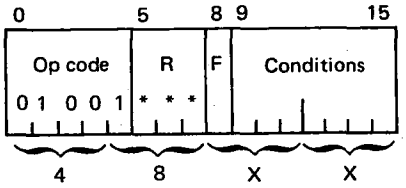
If this instruction is issued on a 5010 Processor Module model A or B, it is treated as a PBALL instruction. (See the description of the PBALL instruction earlier in this chapter.)

Example: Assume that the BUL instruction is at storage locations 87 and 88.



The IAR contains the effective address of the next instruction to be executed; that is, the instruction branched to, which is at location 1058. The branch to location 1058 is taken and execution begins.

Skip Conditional (PSKC)



The skip conditional instruction (PSKC) skips the one-word instruction that immediately follows the PSKC instruction only if certain conditions are present. These conditions are specified within the PSKC instruction. Since only one word is skipped, the PSKC instruction should never be followed by a long-format instruction. Any combination of testing can be requested.

The programmer determines what tests are conducted to decide whether a skip will be taken. This is accomplished by setting bits on or off in the condition field (bits 8 to 15) of the instruction.

The bits and their meanings are shown below:

Bit Set On	*Condition Tested*
9	Carry and overflow indicators both off
10	Zero-result indicator on
11	Negative-result indicator on
12	Positive-result indicator on
13	Even-result indicator on
14	Carry indicator off
15	Overflow indicator off

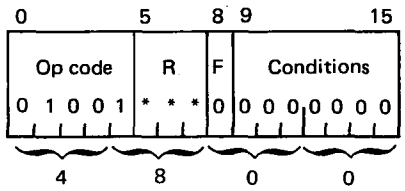
If any one of the specified test conditions is present, the skip takes place by incrementing the IAR by 1 before the next instruction is fetched from storage. If none of the specified test conditions are present, or no conditions are tested (all condition field bits are set to 0's), the instruction sequence does not skip the next word, but rather executes the instruction contained in the next word.

The R field is ignored in this instruction.

The PSKC instruction does not change the carry and result indicators. Testing the overflow indicator with bit 15 always set the overflow indicator off unless condition field bit 8 (overflow-save flag) of the instruction is set on. Testing the overflow indicator with bit 9 does not change the overflow indicator; the overflow-save flag has no effect.

Example 1:

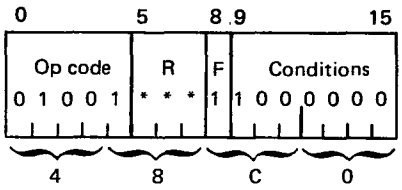
PSKC instruction



No testing is requested since all condition field bits are 0's. No skip occurs.

Example 2:

PSKC instruction



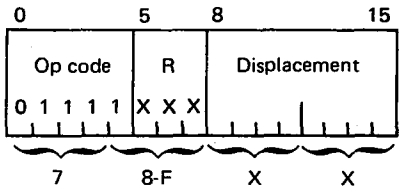
Carry indicator 0

Overflow indicator 1

The only test requested (bit 9 set on) is that both the carry and overflow indicators are off. Since the overflow indicator is on, the test is false and no skip occurs.

Example 3: In example 2, if the overflow indicator is off, the test requested is true and the next word is skipped.

Add to Storage and Skip (PAS)



The add to storage and skip instruction (PAS) increments by 1 the contents of the storage location specified by the effective address. If the result in the storage location is a 0 value, the one-word instruction that follows the PAS instruction is skipped. Since only one word is skipped, this instruction should never be followed by a long-format instruction.

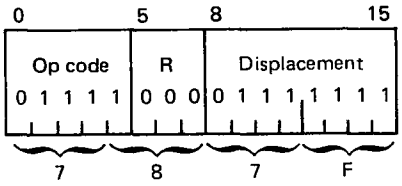
The effective address is the algebraic sum of the contents of the displacement field and the register specified by R. If R=000, the IAR contents are used in this calculation.

The carry and overflow indicators are not changed. The result indicators are changed to reflect the final contents of the addressed storage location.

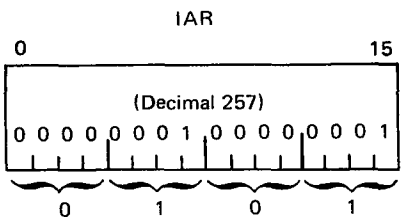
The PAS instruction can be used in conjunction with a storage word that is set up as a counter. Such counters are frequently set up and incremented each time a particular job, function, or routine is accomplished. The counter value changing to 0 can be the condition for termination of the routine and/or branching to another portion of the program. The add to storage and skip instruction (PAS) is used to increment the counter and then skip a storage word when the counter value becomes 0.

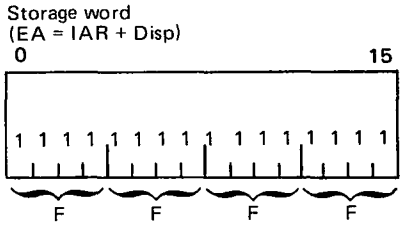
Example:

PAS instruction
(located at address 256 decimal)



PAS instruction
is fetched

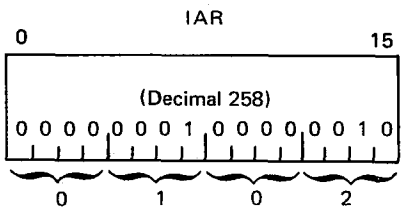


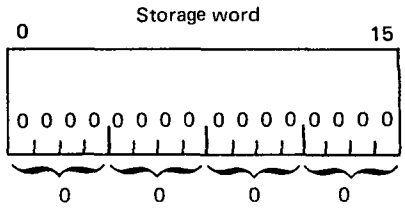


PAS instruction
execution

Contents of storage word	1111 1111 1111 1111
Add one	+0000 0000 0000 0001
Result	0000 0000 0000 0000

After PAS
instruction

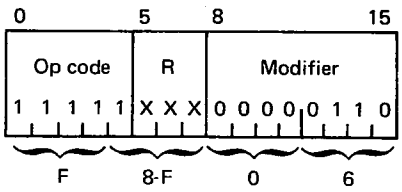




The result-zero and result-even indicators are set on because of the 0 value. The PAS instruction tests the result-zero indicator, finds it on, and thus increments the IAR by 1.

Upon completion of the PAS operation, the next instruction fetched and executed is at location 258. The instruction at location 257 is skipped.

Store to Register (PSTR)



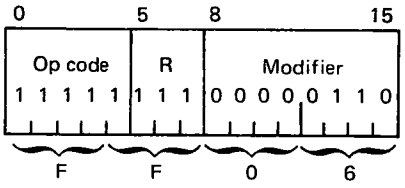
The accumulator contents replace the contents of the register specified by the R field. The contents of the accumulator are not changed.

If R=000, the accumulator contents replace those in the IAR, thus providing a branch to the address specified by the accumulator contents.

The PSTR instruction does not change the carry and overflow indicators. The result indicators are changed to reflect the final contents of the register specified by R.

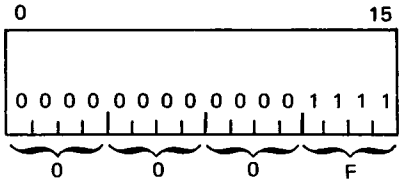
Example:

PSTR instruction

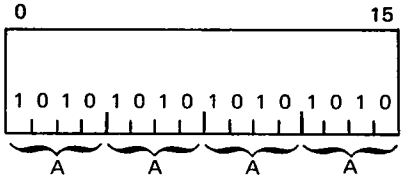


A Before PSTR operation

Accumulator

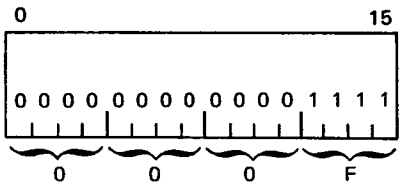


Index register 7

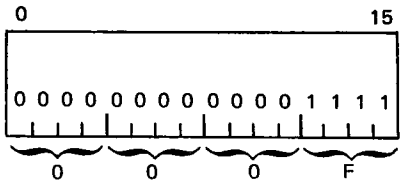


B After PSTR operation

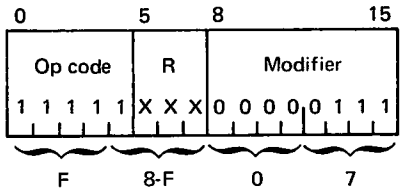
Accumulator



Index register 7



Load from Register (PLR)

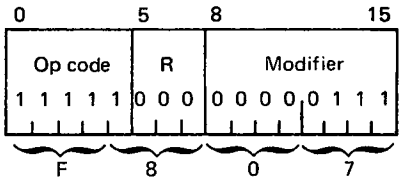


The contents of the register specified by the R field replace the accumulator contents. The contents of the register are not changed. If R=000, the contents of the IAR replace the contents of the accumulator.

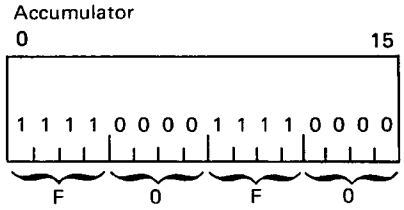
The PLR instruction does not change the carry and overflow indicators. The result indicators are changed to reflect the final contents of the accumulator.

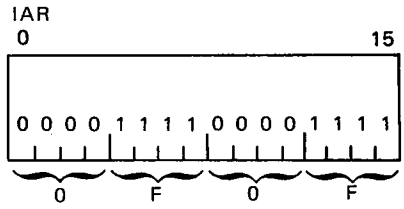
Example:

PLR instruction

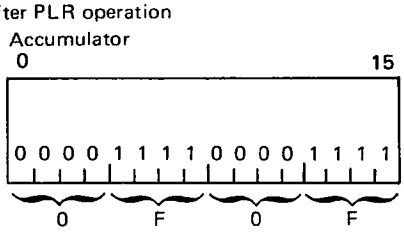


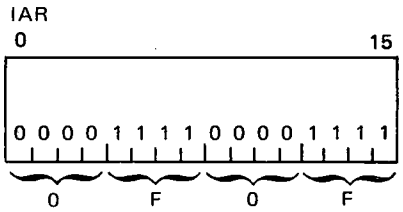
A Before PLR operation



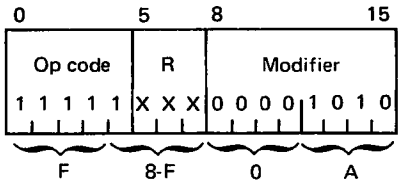


B After PLR operation





Interchange Register (PIR)

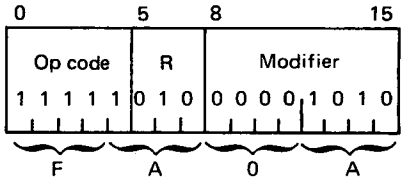


The contents of the accumulator are interchanged with the contents of the register specified by the R field. The contents of the accumulator are placed in the specified register, and the initial contents of the register are placed in the accumulator. Though the carry and overflow indicators do not change, the result indicators are changed to reflect the final contents of the accumulator.

If R=000, the accumulator is interchanged with itself. This results in a no-operation because the contents of the accumulator are interchanged with themselves and, effectively, no change takes place. In this case, the result indicators are not changed.

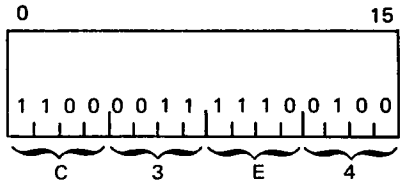
Example:

PIR instruction

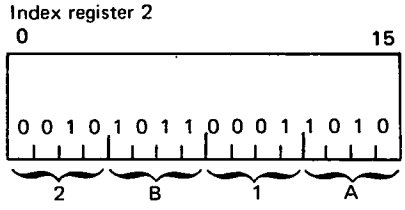


A Before PIR operation

Accumulator

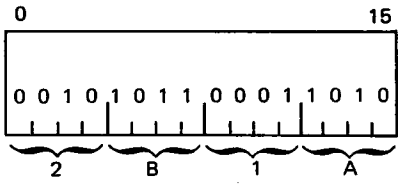


Index register 2

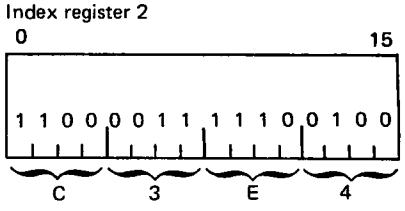


B After PIR operation

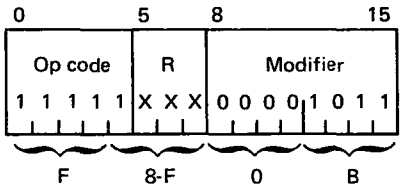
Accumulator



Index register 2



Load Processor Status (PLPS)



Sixteen bits of processor status information are stored into the index register (or accumulator if R=000) specified by the R field. Processor status information is represented by the values of individual bit positions in the word stored in the register.

The bit positions and their meanings follow.

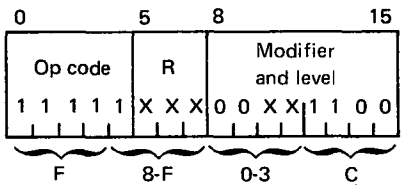
Bit Set On	*Meaning*
0	Invalid shift count
1	Invalid storage address
2	Invalid op-code, function, or modifier
3	Local storage parity check (XR parity check)
4	SDR parity check
5	Control check
6	I/O check
7	Sequence indicator
8	Power warning–master unit
9	Thermal warning–master unit
10	Power warning–remote unit
11	Thermal warning–remote unit
12–15	(Not used, but are 0's)

Bits 0 to 2 are set by program check error conditions, bits 3 to 6 are set by machine check error conditions. Bit 7 is set on by execution of a supervisor call (SVC) instruction and remains on until execution of a PLPS instruction. Bits 8 to 11 are set by a power/thermal warning. (For more information on these conditions, see "Class Interruptions" in Chapter 3.)

Execution of this instruction resets status indicator bits 0 to 7; indicator bits 8 to 11 are reset only when the cause of the warning disappears. None of these status indicators are reset by the reset generated by a machine check interruption.

The PLPS instruction does not change the carry and overflow indicators. The result indicators are changed to reflect the contents of the R register.

Inspect IAR Backup (PIIB)



The contents of a backup IAR (associated with a particular priority level) are stored into the index register (or accumulator if R=000) specified by the R field in the instruction. The contents of the selected IAR are not changed.

The IAR priority level is selected by the binary encoded value in bits 10 and 11 of the instruction as follows:

Bits		*Priority Level*
10	*11*	
0	0	0
0	1	1
1	0	2
1	1	3

Since there is no backup IAR for priority level 0, a 0 value is stored in the R register when bits 10 and 11 select priority level 0.

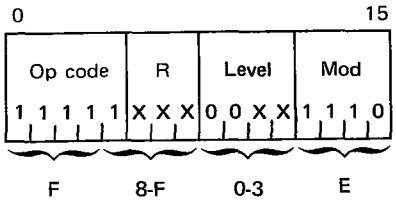
If the PIIB instruction selects the backup IAR for the current priority level, the processor exits from the level and the level is reset.

If the PIIB instruction selects the backup IAR for a level that has been interrupted and is still pending, the carry indicator for the current level will be turned on and the selected level will be reset (it will no longer be pending).

If the PIIB instruction selects the backup IAR for a level that is neither currently active nor pending, the carry indicator for the current level will be turned off.

The PIIB instruction does not change the overflow indicator. The result indicators are changed to reflect the contents of the R register.

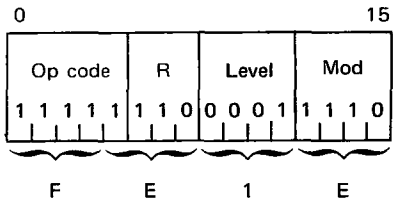
Write IAR Backup (WIB) (5010 Processor Module Model E only)



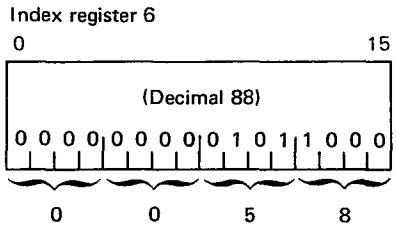
The contents of the register on the current level specified by the R field, or the accumulator if R = 000, replace the contents of the instruction address register backup on the selected level. The level is selected by the binary encoded value in bits 8 through 11 of the instruction. The contents of the register specified by the R field are not changed. Since level 0 has no instruction address register backup, no-operation is performed if level 0 is selected. The carry, overflow, and result indicators are not changed.

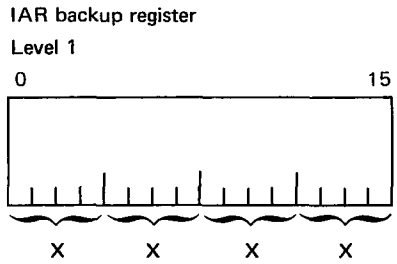
Example:

WIB instruction

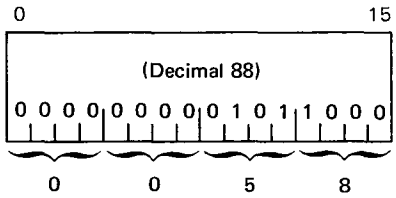


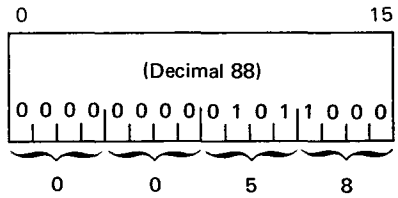
WIB instruction is fetched



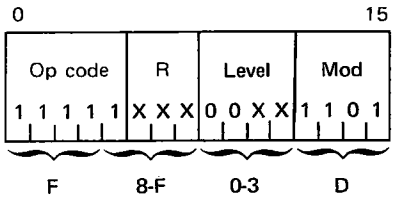


After WIB instruction execution





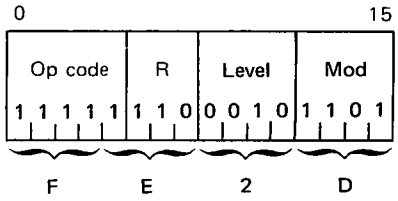
Read IAR Backup (RIB) (5010 Processor Module Model E only)



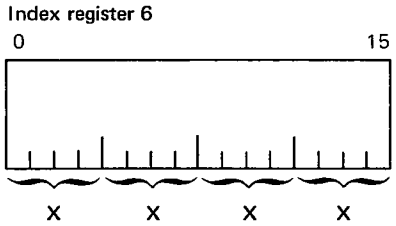
The contents of the instruction address register backup for the selected level replace the contents of the register on the current level, specified by the R field, or the accumulator if R = 000. The level is selected by the binary encoded value in bits 8 through 11 of the instruction. Since level 0 has no instruction address register backup, this instruction results in the specified register having a 0 value if level 0 is selected. The carry, overflow, and result indicators on the selected level are not changed. The result indicators on the current level are changed depending on the contents stored in the specified register.

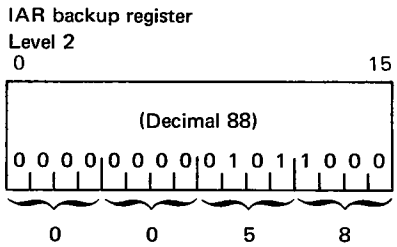
Example:

RIB instruction

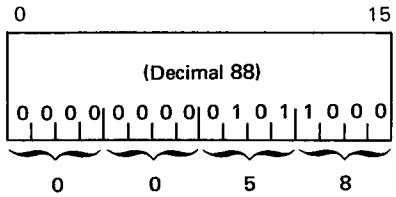


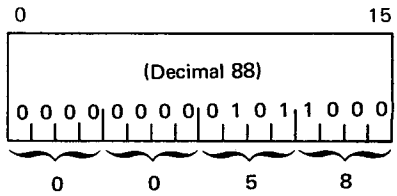
RIB instruction is fetched



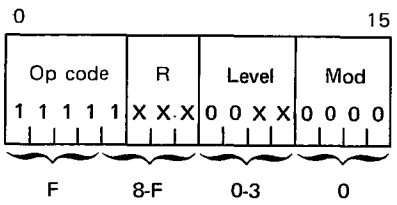


After RIB instruction execution





Store Indicators (STI) (5010 Processor Module Model E only)



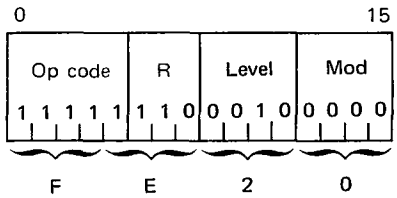
The contents of the result, carry, and overflow indicators for the selected level replace the contents of the register, on the current level, specified by the R field, or the accumulator if R = 000. The level is selected by the binary encoded value in bits 8 through 11 of the instruction. The carry, overflow, and result indicators on the selected level are not changed. The result indicators on the current level are changed depending on the contents stored in the specified register. The carry and overflow indicators on the current level are not changed.

Status Register Contents

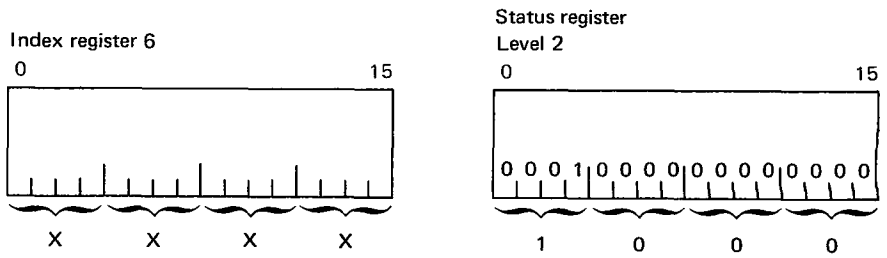
Bit	*Contents*
0	Reserved
1	Reserved
2	Zero result indicator
3	Negative result indicator
4	Positive result indicator
5	Even result indicator
6	Carry indicator
7	Overflow indicator
8–15	Reserved

Example:

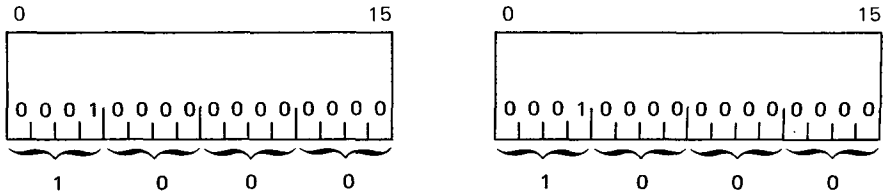
STI instruction



STI instruction is fetched

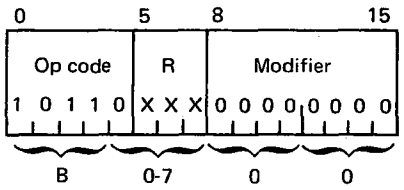


After STI instruction execution



Only bit 3 is on, so the negative result indicator is the only program indicator set on for level 2.

AND to Mask (PNM)



The 4-bit interruption mask register is ANDed, bit by bit, with bits 0 to 3 of the index register (R), or the accumulator if R=000. This result replaces the contents of the interruption mask register, where a bit value of 1 means that an interruption is permitted on a particular priority level. The contents of the R register are not changed; neither are the carry, overflow, and result indicators.

Executing the AND-to-mask instruction turns off the summary mask function. To do so without changing the IMR, the R register must have bits 0–3 set to 1111.

Interruption requests are not recognized by the processor during execution of the PNM instruction. After execution of the PNM instruction, interruption control is returned to the mask register, and samples for power/thermal warning class interruptions and priority interruptions are resumed.

ANDing occurs only between corresponding bits in the interruption mask register and the R register: bit 0 is ANDed only with bit 0, bit 1 only with bit 1, and so on. The four possible ANDing combinations are:

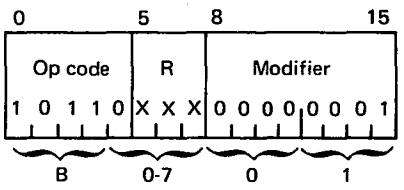
Bit Value from R Register	*Bit Value from Interruption Mask Register*	*Result in Interruption Mask Register*
0	0	0
0	1	0
1	0	0
1	1	1

Thus, a bit value of 1 will result in the mask register only if a corresponding bit position is set to a value of 1 in both the mask register *and* the R register.

Example:

Register bits 0 to 3	1101	
Interruption mask register	0111	(interruptions permitted on levels 1, 2, and 3)
Result in interruption mask register	0101	(interruptions permitted on levels 1 and 3)

OR to Mask (POM)



The 4-bit interruption mask register is ORed, bit by bit, with bits 0 to 3 of the index register (R), or the accumulator if R=000. This result replaces the contents of the interruption mask register, where a bit value of 1 means that an interruption is permitted on a particular priority level. The contents of the R register are not changed; neither are the carry, overflow, and result indicators.

Executing the OR-to-mask instruction turns off the summary mask function. To do so without changing the IMR, the R register must have bits 0–3 set to 0000.

Interruption requests are not recognized by the processor during execution of the POM instruction. After execution of the POM instruction, interruption control is returned to the mask register, and samples for power/thermal warning class interruptions and priority interruptions are resumed.

ORing occurs only between corresponding bits in the interruption mask register and the R register: bit 0 is ORed only with bit 0, bit 1 only with bit 1, and so on. The four possible ORing combinations are:

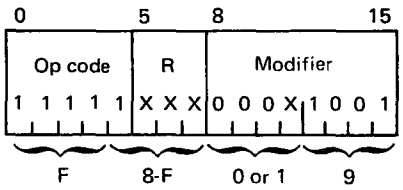
Bit Value from R Register	*Bit Value from Interruption Mask Register*	*Result in Interruption Mask Register*
0	0	0
0	1	1
1	0	1
1	1	1

Thus, a bit value of 1 will result in the mask register if a corresponding bit position is set to a value of 1 in the mask register and/or the R register.

Example:

Register bits 0 to 3	1101	
Interruption mask register	0101	(interruptions permitted on levels 1 and 3)
Result in interruption mask register	1101	(interruptions permitted on levels 0, 1, and 3)

Sense Level and Mask (PSLM)

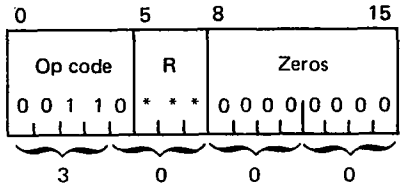


The currently active priority level number and the contents of the interruption mask register are stored in the index register (R), or the accumulator if R=000. The priority level is stored as a binary number in bits 14 and 15 of the register; the interruption mask is stored in bits 0 to 3 exactly as it appears in the interruption mask register. Bits 4 to 13 are set to 0's.

The PSLM instruction does not change the carry and overflow indicators, or the IMR. The result indicators are changed, however, to reflect the final contents of the R register.

The summary mask can be turned on by setting modifier bit 11 in the PSLM instruction to a value of 1. No priority or power/thermal warning interruptions can occur until the summary mask is turned off. A system reset has somewhat the opposite effect: it sets on all four bits of the interruption mask register so that I/O interruptions can occur on any level.

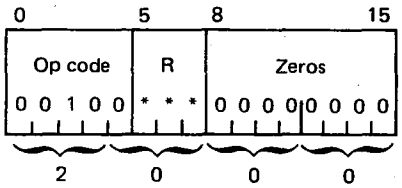
Level Exit (PLEX)



The processor exits the current priority interruption level. If lower-level interruptions are pending, the system remains active and services the request on the highest pending priority level. If no other interruptions are pending, the processor enters the wait state.

The PLEX instruction does not change the carry, overflow, and result indicators.

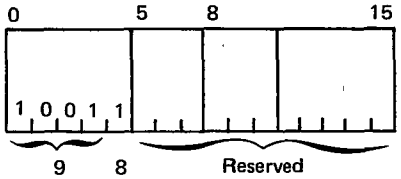
Stop (PSTP)



The processor enters the stop state, provided the System/7 console controls switch is in the enable position. If this switch is not in the enable position, the PSTP instruction performs as a no-operation instruction.

The PSTP instruction does not change the carry, overflow, and result indicators.

Supervisor Call (SVC)



The summary mask is turned on and a branch is taken via the contents of main storage location hex 0009. The sequence indicator in the processor status word is turned on. No other indicators are changed. Power/thermal warning interruptions are inhibited during this instruction. The contents of the instruction address register (IAR) are stored ~~in the~~ address specified by the interruption level transfer vector (ILTV) for the level that is executing. For example, if the SVC is issued on level 3, the contents of the IAR (the IAR points to the location of the SVC) are stored in the address specified by the contents of main storage location hex 0013.

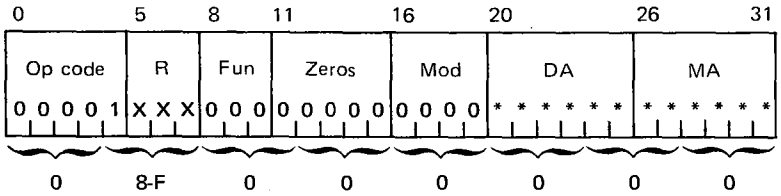
To allow the system to respond to as many error conditions as possible, the load processor status (PLPS) instruction must be executed after the SVC instruction. If not, a machine check or program check condition causes a machine check class interruption. For example, assume that an SVC instruction has been issued and a PLPS has not been executed. If a program check condition occurs or another SVC is fetched from main storage, the processor status word (PSW) is first cleared, then set with the appropriate program check condition along with control check. If a second SVC is being fetched, a program check condition is set. The setting of control check causes a machine check class interruption if the keyswitch is in the disable position or the check control switch is in

the process position. If the system is in stop on error mode, an error halt occurs. If a machine check condition occurs, the PSW is first cleared, then set with the appropriate machine check condition. If a power/thermal warning condition occurs, and the summary mask has been turned off by a PNM or POM instruction prior to the PLPS instruction, a power/thermal warning class interruption occurs.

Because of the possible occurrence of these conditions, it is important that the summary mask is not turned off between the execution of an SVC and the PLPS instruction.

STORAGE PROTECT INSTRUCTIONS (5010 PROCESSOR MODULE MODEL E ONLY)

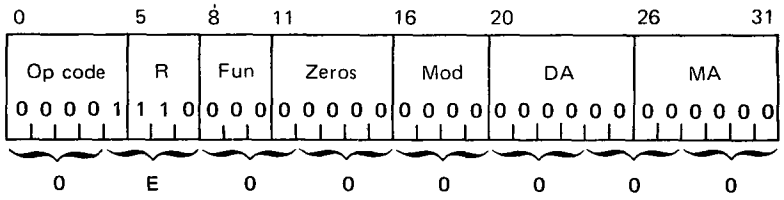
Alter Storage Key (ASK)



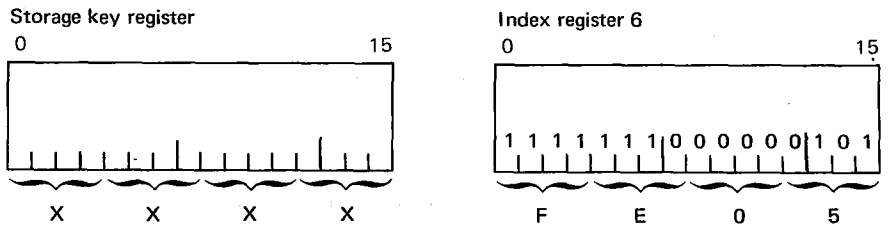
Bits 0 through 6 of a register, or the accumulator if R = 000, select the storage segment. These bits are the seven high-order bits that are common to all 512 addresses in the storage segment. Bits 13 through 15 are transferred into the storage key register for the selected storage segment. This instruction enables the storage protect mechanism, which if previously enabled remains enabled. The result indicators are not changed but the carry and overflow indicators are reset to 0.

Example:

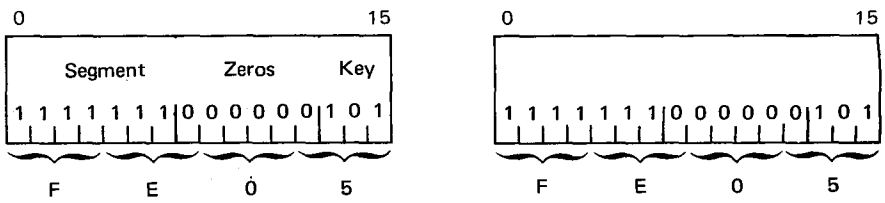
ASK instruction



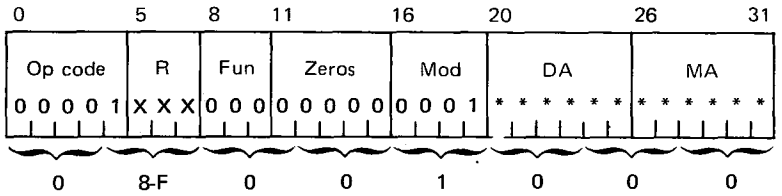
ASK instruction is fetched



After ASK instruction execution



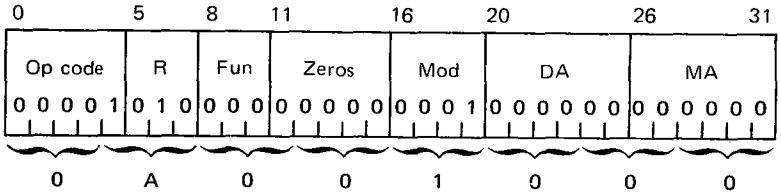
Alter Protect Key (APK)



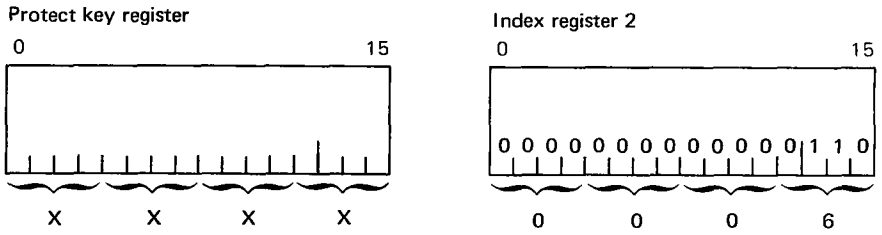
Bits 13 through 15 of a register, or the accumulator if R = 000, are loaded into the protect key for the current level. This instruction also enables the storage protect mechanism, which if previously enabled remains enabled. The result indicators are not changed but the carry and overflow indicators are reset to 0.

Example:

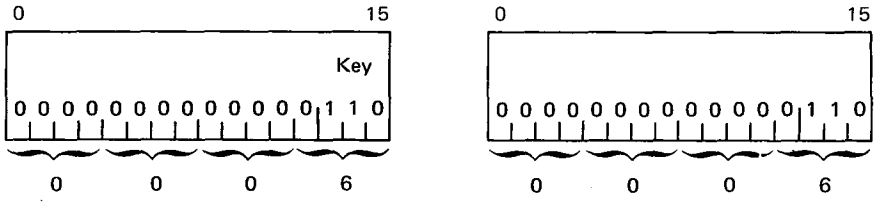
APK instruction



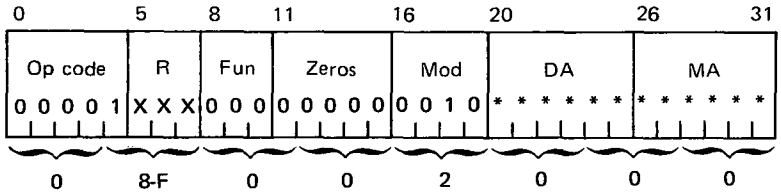
APK instruction is fetched



After APK instruction execution



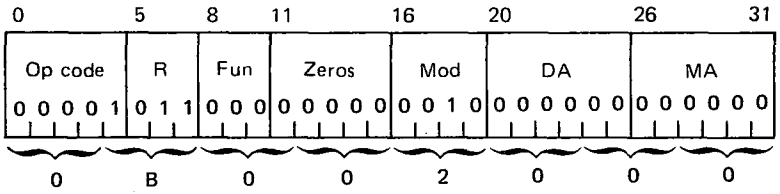
Read Protect Key (RPK)



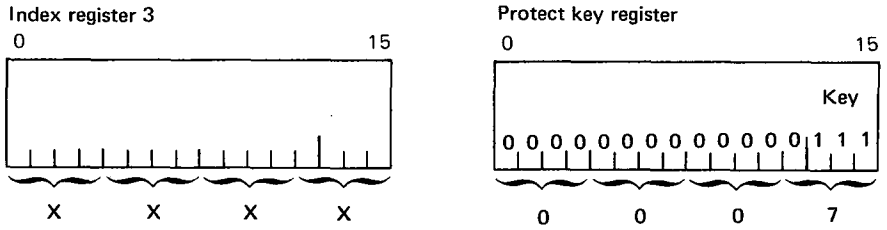
Bits 13 through 15 of a register, or the accumulator if R = 000, are replaced by the contents of the protect key. The result indicators are changed to reflect the value loaded into the register. The carry and overflow indicators are reset to 0.

Example:

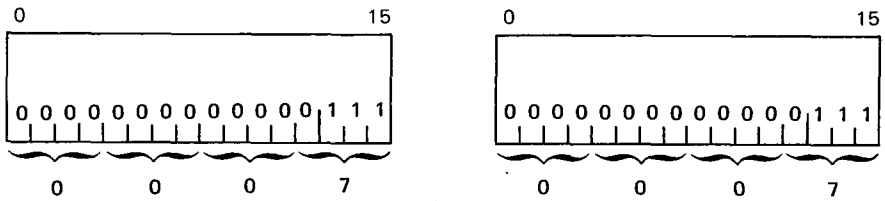
RPK instruction



RPK instruction is fetched



After RPK instruction execution

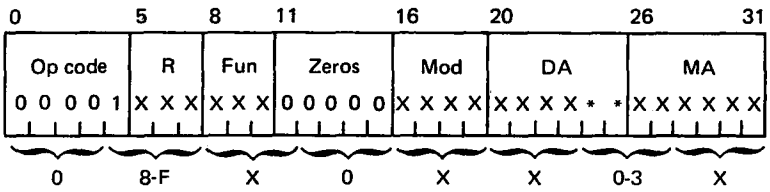


INPUT/OUTPUT INSTRUCTION

One instruction operation code services I/O devices. Function bits in the input/output instruction determine which one of the following basic I/O commands is performed:

1. Immediate Write. Sends one word of data from a processor register to an I/O device.
2. Immediate Read. Receives one word of data from an I/O device and places the data into a processor register.
3. Prepare I/O. Sends one word of interruption control information from a processor register to an I/O device.
4. Halt I/O. Resets an I/O device.
5. Set Interrupt. Sends one word of data to establish an interruption request, either on a priority level or to a host processor.
6. Read Cycle Steal. Sends up to 3,072 words of data from the disk storage module to main storage.
7. Write Cycle Steal. Sends up to 3,072 words of data from main storage to the disk storage module.

Execute I/O (PIO or XIO)



One of the seven basic input/output commands is performed, using an index register (or the accumulator if R=000) if data or control information is needed.

The function (Fun) field (bits 8 to 10) determines the type of I/O operation to be performed. Valid values and their meanings are:

001—Immediate write
010—Immediate read
011—Prepare I/O
100—Halt I/O
101—Set interrupt
110—Read cycle steal
111—Write cycle steal

A function field value of 000 causes a program check interruption on processor module models A and B. The seven valid I/O operations are discussed in greater detail under "I/O Command Functions."

Bits 11 to 15 of the PIO instruction are not used, but their values must be set to 0.

The modifier field, bits 16 to 19 (bits 0 to 3 of the second instruction word), is used only with the read or write function code. These modifier bits further define the read or write operation to be performed. The contents of the modifier field are indicated in the format of each specific I/O command when the command is described for an I/O device or module.

The device address field, bits 20 to 25 (bits 4 to 9 of the second instruction word), specifies a point, group, or device within a module. Bits 20 to 23 contain a 4-bit subaddress; bits 24 and 25 are reserved and must be set to 0. The assigned device addresses are discussed separately for each module.

The module address field, bits 26 to 31 (bits 10 to 15 of the second instruction word), selects the 5026 Enclosure and the I/O module location being addressed by the PIO instruction as shown in Figure 4-3.

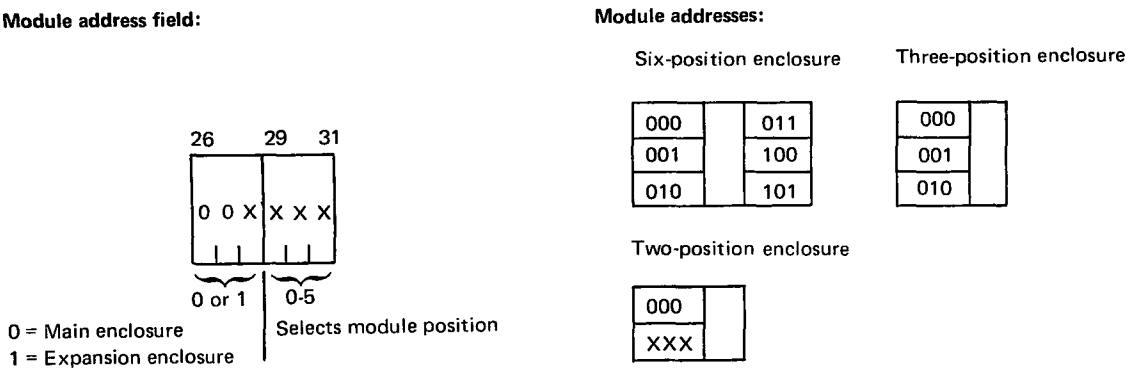


Figure 4-3. System/7 module addressing

With the two-position enclosure, any value in bits 29 to 31 (bits 13 to 15 of the second instruction word) other than 000 addresses the single I/O module position. The value chosen to initially address the single I/O module must also be used for all subsequent addressing of that module by the same program. However, because of the possibility of future expansion of such a configuration, it is advisable to specify a value of 010 or 101. By specifying one of these two values, the I/O module can be installed in the same relative position in a larger configuration (that is, at the bottom of the enclosure). Thus, the amount of program redesign and customer termination rewiring will be minimized when changing to a three- or six-position enclosure.

The processor module occupies the first position in the main enclosure; therefore, its module address is always 000000.

Each time an I/O command is issued, the processor determines whether (1) the command was executed successfully, (2) an error occurred, (3) the device was busy or an interruption was pending, or (4) the device was not attached to the system. One of these machine conditions is indicated by the two-bit condition code, whose settings are determined (and made available to the program) by the results of the I/O operation. The two bits used for condition codes are the same two bits that serve as carry and overflow indicators for other machine instructions. A summary of the condition codes and their settings as indicated by the carry and overflow indicators is shown in Figure 4-4.

The result indicators are changed only by an immediate read command to reflect the final contents of the register specified by the R field of the PIO instruction.

Condition code	Carry (C) / Overflow (O)	Command functions				
		Write	Read	Prepare I/O	Halt I/O	Set interrupt
0	C = 0 / O = 0	S	S	S	S	S
1	C = 0 / O = 1	E	E+	E		
2	C = 1 / O = 0	B	B*+	B		B
3	C = 1 / O = 1	X	X+	X	X	X**

Key to symbols:
S = Satisfactory operation
E = Error detected
B = Busy or interruption pending
X = Device not attached

* = Not used on read status command
** = Returned only if set interrupt is directed at host and no host processor is attached.
\+ = With immediate read command, validity of target register contents is not guaranteed. Register may have invalid data and/or parity.

Figure 4-4. Condition codes

Immediate Write (Fun=001)

An immediate write command transfers one word (16 bits) of data from a specified processor register to the addressed device. The result indicators are not changed.

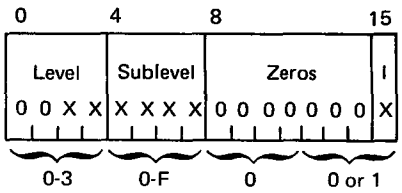
Immediate Read (Fun=010)

An immediate read command transfers one word (16 bits) of data from the addressed device to a specified processor register. The result indicators are changed to reflect the final contents of the specified register.

Prepare I/O (Fun=011)

The prepare I/O command transfers one word (16 bits) of data from a specified processor register to the addressed device. The difference between this command and the immediate write command is the data and its purpose. The immediate write command transfers data for transmission to the output device or point, or for control of attached I/O devices. The prepare I/O command transfers information advising the device whether it is allowed to interrupt and, if so, on what priority level and sublevel.

The data sent to the device by the prepare I/O command has the following format and meaning:



The significance of the data fields in this word are described as follows.

Level: This 4-bit field specifies the priority interruption level assigned to the interrupting source. The binary value of bits 2 and 3 indicates the priority level (0, 1, 2, or 3). Bits 0 and 1 must be 0.

Sublevel: This 4-bit field assigns a sublevel (from 0000 to 1111) to the interrupting source. When requesting an interruption, the source presents this sublevel so that the system can locate the starting address of the servicing routine in the appropriate level table. A sublevel of 0000 accesses the second word in the table; a sublevel of 1111 accesses the seventeenth (last) word. The first word in each level table is reserved for the old instruction address associated with a program check interruption. (Refer to Chapter 3, "System/7 Interruptions," for more detail.)

Zeros: This field is not used, but its value must be set to 0.

I: This bit determines whether the device is allowed to request an interruption. An I-bit value of 1 permits the device to request an interrupt, and a value of 0 prevents it. (The interruption mask determines whether the device can actually interrupt the processor.)

If the device is not permitted to cause an interruption, issuing an interruption-causing command results in:

1. Condition code 1 being set.
2. The command reject indicator being set in the DSW.
3. The command not being executed.

The prepared device stores the PIO data and presents the priority level and sublevel to the processor each time the device presents an interruption request (and interruptions are enabled). Data stored at the device is reset on a system reset, a power-on reset, or changed by the successful execution of another prepare I/O command to the device.

Halt I/O (Fun=100)

The halt I/O command resets the addressed I/O module or control, with the exception of sensor-based output points and interruption information sent by prepare I/O commands. All other controls, device status, and pending interruptions are reset. The R field is ignored.

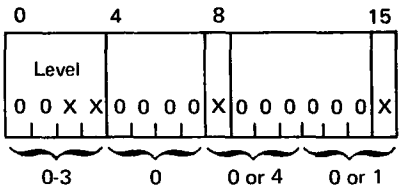
Halt I/O to the processor module (module address = 000000 in the instruction) causes a program check interruption.

If the device has an outstanding request in the interruption buffer, that request is *not* reset by halt I/O. Therefore, an interruption request can be honored by the system even though the interrupting source has been reset.

Set Interrupt (Fun=101)

The set interrupt command is issued to the processor module. One word of data is sent to the processor interruption controls directing them to establish an interruption request either on a specified priority level or to an attached 1130 processor. Both the device address and module address fields in the set interrupt PIO command must be all 0's, signifying that the processor module is addressed. An interruption request directed to a priority level in the System/7 presents a sublevel of 0 to the processor when the request is serviced.

The data word transmitted from a processor register to the interruption buffer has the following format:



Bits 4 to 7 are 0's.
Bit 8 is set according to the requirements of the 5024 I/O Attachment Enclosure. Otherwise it is zero.
Bits 9 to 14 are 0's.
Bit 15 is set according to the following two sections and Chapter 17.

Request Interrupt to System/7: Bit 15 set to 0 requests that an interruption request be directed to the System/7 only. The interruption request is established in the buffer reserved for the priority level indicated by the level field contents in the data word.

Two buffers are used for each interruption level: one for general interruptions and one especially for set interruptions. If the general interruption buffer is available, the set interrupt request enters the buffer and is recognized the same as any other interruption request. If the general interruption buffer is full, the set interrupt request enters the special set interruption buffer. When the general interruption buffer is again available, the set interrupt request enters the general buffer in competition with any other request attempting to enter the general buffer for that level. In any case, when the set interrupt request is recognized by the processor, the request presents a sublevel and device address of 0's and a module address of 000111.

If a set interrupt request is pending in either buffer, condition code 2 is returned. If no set interrupt request is pending, the set interrupt request enters the general interruption buffer (if it is free) or the special set interruption buffer.

Condition code settings for this use of the set interrupt command are:

00—An interruption request was successfully entered into the appropriate level buffer.
01—Does not occur with this use of the command.
10—A set interrupt request is already pending on the specified level.
11—Does not occur with this use of the command.

Request Interrupt to 1130: Bit 15 set to 1 in the data word sent by the set interrupt PIO command requests that an attention interruption request be established to the 1130 processor. The contents of the level field in the data word are ignored.

The 1130 host attachment in the System/7 processor module generates an interruption on the 1130 SAC, and sets on the attention status bit for use by the 1130 system when it services the interruption.

Condition code settings for this use of the set interrupt command are:

00—An interruption request was made and attention status bit was set on in the 1130 host attachment (does not indicate whether the 1130 has received and/or recognized the interrupt).
01—Does not occur with this use of the command.
10—An attention status bit was already set on, so this command is not honored.
11—1130 processor is off line or not in the System/7 configuration.

5024 I/O Attachment Enclosure. For information see Chapter 17.

Read Cycle Steal (Fun=110)

A read cycle steal command transfers up to 3,072 words of data from the disk storage module to main storage. The result indicators are not changed by this command.

Write Cycle Steal (Fun=111)

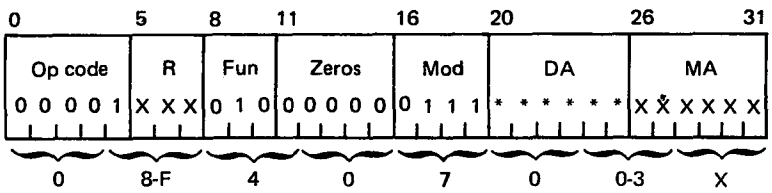
A write cycle steal command transfers up to 3.072 words of data from main storage to the disk storage module. The result indicators are not changed by this command.

Status Indicators

Status bits are used in the System/7 to indicate to the program the status of I/O devices attached to the system. These status bits can represent error conditions or they can represent normal operating conditions. Some status bits also present an interruption request to the processor. There is one device status word (DSW) associated with each module in the System/7. An interrupt status word (ISW) is associated with each interrupting source within a module, since some modules can contain several types of I/O devices. (For example, the operator station, timer, ACCA, and BSCA circuits all reside in the processor module.) A status condition may set multiple bits in the ISW for that device.

Device Status Word (DSW)

The 16 bits in the DSW are set on as a result of errors occurring during the execution of an I/O instruction, that is, before the condition code is set. No interruptions result from setting on a DSW bit because the presence of recorded errors is indicated to the program by condition code 1. The DSW is read by the read DSW command in the following general format:



The read DSW command and the specific status bits that apply are described for each I/O module in its respective chapter of this manual.

Once an error is recorded in the DSW, any subsequent command (except halt I/O and read DSW) to the device is rejected and the condition code is set to 1 until the DSW is reset by a system reset, a halt I/O, or a read DSW command.

A general DSW has the following format, with modules providing the bits pertinent to their operation. (Bits not used by the module are set to 0.)

Significant Bits	*Meaning*
1	Command reject. The addressed module cannot execute the command issued. An example is an interruption-causing command issued to a device disabled for interruptions.
3–7	Device dependent. (See individual devices.)
14	Data check. A parity error involving data is detected on the interface. The operation that caused this condition can usually be retried successfully if the error condition is intermittent. This error bit does not apply to the DSW for the processor module because the machine check interruption performs this function.
15	Invalid device address. This bit is set on if the point, group, or device is not available to the system. (This bit does not apply to all I/O modules.)

Any error encountered during an immediate read DSW or read ISW is recorded in the DSW, replacing the original contents, and setting the condition code to 1.

Interrupt Status Word (ISW)

The 16 bits in the ISW are set when errors are detected after completion of immediate I/O commands and while the device is busy. Normally, any ISW bit set on (except the device-busy bit) causes the device to present an interruption request. Each interrupting device in a module has its own ISW.

An ISW can be read by the immediate read ISW command. If this instruction is successfully executed, the ISW is reset provided its interruption request has been accepted by the system. The ISW is also reset by halt I/O, system reset, or the first new selection of the device by a command (except immediate read ISW) after a pending interruption from the device has been accepted by the system.

The read ISW command is described for each interrupting source in the respective chapter covering that device or module.

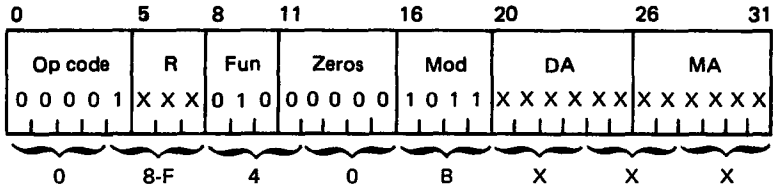
A general ISW has the following format, with the interrupting sources providing the bits pertinent to their operation. (Bits not used by the interrupting source are set to 0.)

Significant Bits	*Meaning*
0	Attention. The device has detected an external condition that is significant to the program. The condition is interpreted by the program and is not associated with the initiation, execution, or termination of an I/O operation. If the device is busy or has an interruption pending, the attention bit will be presented with the interruption that occurs later when the device becomes not busy. Otherwise, an attention condition will cause the device to request an interruption.
2	Overrun. A device requires data servicing beyond the capability of the system to accept data. This bit may or may not be considered an error bit, depending on the use of the bit within the I/O module.
3–7	Device dependent. (See individual devices.)
12	Device busy. The device is working, either busy or waiting for an interruption to occur.
13	Device end. The device has terminated an operation in a normal manner. This is not an error condition.
15	Device dependent. (See individual devices.)

Interrupt Status Word Extension (ISWEX)

Some I/O devices or modules cannot present enough status information in one 16-bit word, so a second status word, called interrupt status word extension (ISWEX) is used. Conditions that set any ISWEX indicator bit also set a bit in the ISW for that device or module.

An ISWEX can be read with the following command:

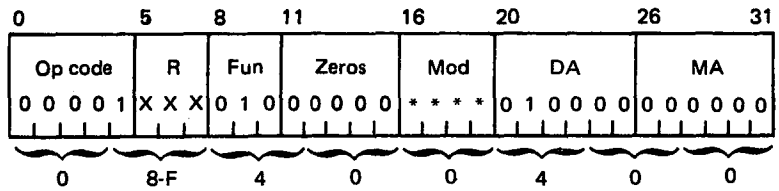


This command stores 16 bits of status information into the index register specified by R, or the accumulator if R=000. Each I/O device that employs an ISWEX determines the condition(s) that set any particular bit. These conditions are defined in the chapter that describes the specific I/O device concerned. (Bits not used by the device are set to 0.)

Direct Control Channel Status Word

Errors may be detected during the execution of immediate commands to devices attached directly to the processor module (for example, timers, operator station, Asynchronous Communications Control attachment or Binary Synchronous Communications Adapter). Such errors will cause certain direct control channel status bits to be set on or off. Errors detected by such commands cause condition code 1 to be returned to the processor module. This condition code is returned to any subsequent I/O commands addressed to the processor module until the status word is reset.

The direct control channel status word is read and reset by an immediate read command similar to the read DSW command as follows:



This command stores the status word in the index register specified by the R field, or the accumulator if R=000.

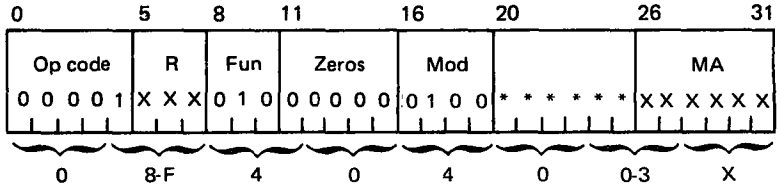
The status word is reset when read by the above read command or by system reset.

Significant bits in the direct control channel status word and their meanings follow. (Bits not used by the direct control channel are set to 0.)

Significant Bits	*Meaning*
1	Command reject. The addressed device has received a command that it is not designed to execute, or that it cannot execute because of the device's present state. Condition code 1 is set and the command is not executed. This command reject bit may also be set in conjunction with setting on the following bits 3 or 4.
3	Operator station check. May be caused by: a. operator station motor not running. b. operator station not being prepared to request interruptions. c. issuing a punch command when there is a low paper tape supply in the punch. d. issuing a command, other than read ISW, and the operator station is not in remote mode. e. issuing a feed or read tape command and • the paper tape reader switch is open, or • the read tape function is not selected, or • no paper tape in the reader, or • paper tape from the punch is tight. This operator station check bit may also be set in conjunction with the setting on of bits 1 or 15.
4	Host attachment check. This bit is set on when an optional host attachment, such as ACCA or BSCA, detects any error condition. This host attachment check bit may also be set in conjunction with the setting on of command reject (bit 1).
15	Invalid device address. An immediate command was issued to the processor module but the device address is for a device not available in the processor module; that is, not installed or off-line. The operator station in the local mode of operation is considered off-line.

I/O Module Identification Word

All I/O commands issued to I/O modules specify a module address. This address denotes a positional address within the 5026 Enclosure(s), and can vary from 00 to 05 or from 08 to 0D hexadecimal. Any module except the processor module can be relocated within the 5026 Enclosure(s). Thus, only the module address need be changed in the I/O commands intended for a particular I/O module. Each I/O module presents an identification word when the module is addressed by the following read-ID command:



Sixteen bits of module identification data from the addressed I/O module are stored into the index register (R), or the accumulator if R=000.

The format for the data word is shown in Figure 4-5. Bits 1 to 7 identify the type of I/O module installed. Bits 8 to 15 identify the devices or features installed within the module.

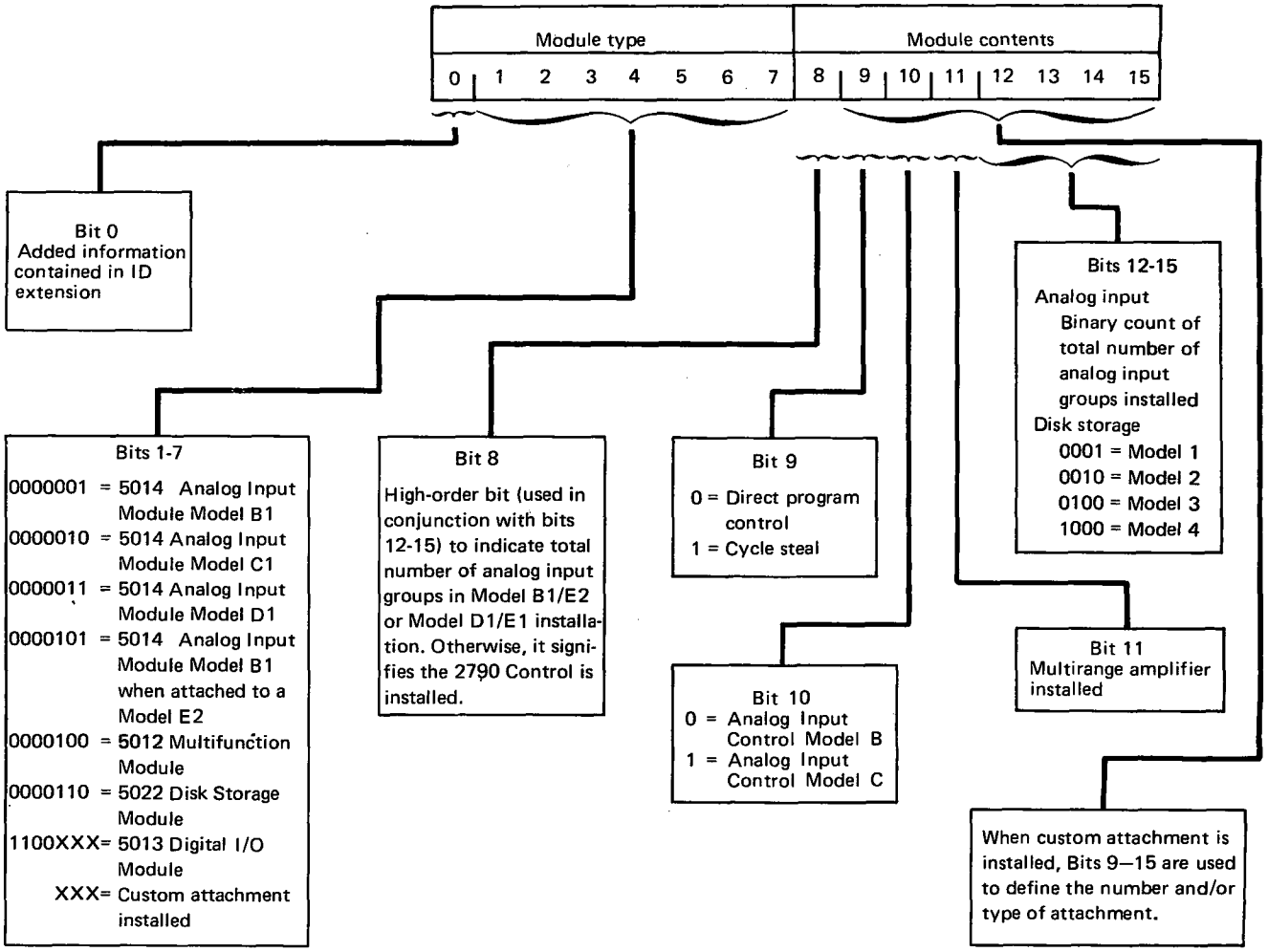


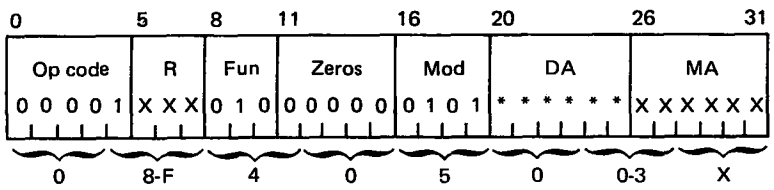
Figure 4-5. Module ID word

Analog input groups must be installed sequentially; thus, ID bits 12 to 15 = 0011 indicate that groups 1, 2, and 3 are installed.

When the digital I/O module with custom attachment is installed, bits 9–15 identify the number and type.

Some I/O modules cannot present enough identification data in one 16-bit word, so a second word, called ID extension word, is required.

Bit 0 set on in the ID word indicates that additional information should be obtained from an ID extension word by the following command:

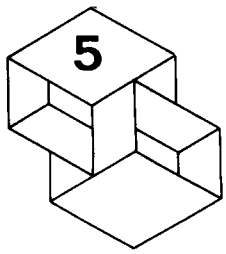


Sixteen bits of additional module identifying data are stored into the index register (R), or accumulator if R=000. Each bit in the data word is set on or off to indicate the presence or absence of input or output groups.

The following list indicates the meaning of each bit when it is set on.

Bit	*Meaning*
0	Digital input group 1 has interrupt feature
1	Digital input group 0 has interrupt feature
2	Digital output group 3 is present
3	Digital output group 2 is present
4	Digital output group 1 is present
5	Digital output group 0 is present
6	Analog output point 1 is present
7	Analog output point 0 is present
8	Digital input group 7 is present
9	Digital input group 6 is present
10	Digital input group 5 is present
11	Digital input group 4 is present
12	Digital input group 3 is present
13	Digital input group 2 is present
14	Digital input group 1 is present
15	Digital input group 0 is present

CHAPTER CONTENTS



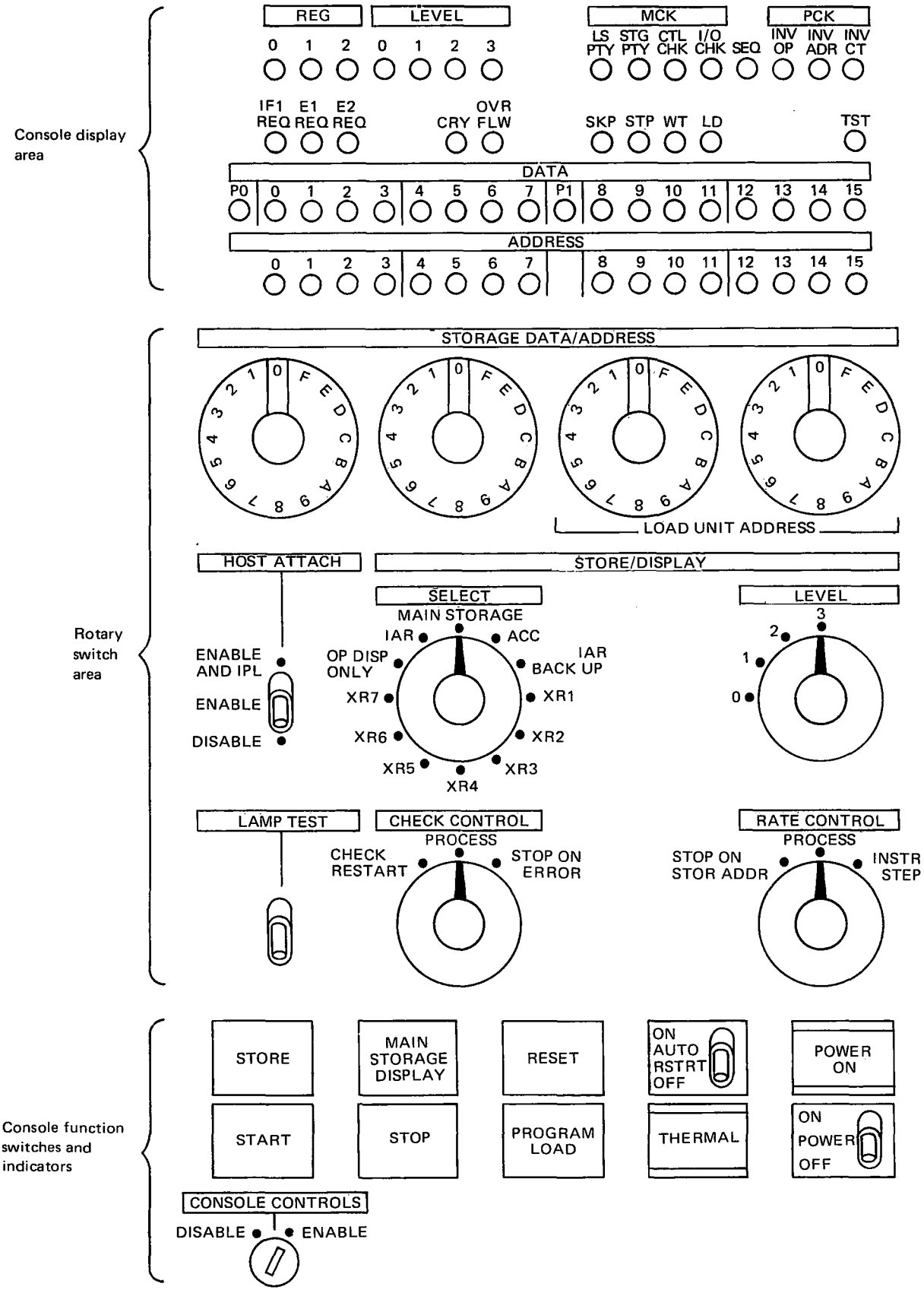


Figure 5-1. Operator console

The switches, keys, and lights needed to operate and control the System/7 are arranged on the operator console of a three- or six-position main enclosure as shown in Figure 5-1. The descriptions in this chapter also apply to the console on the two-position enclosure, even though the switches, keys, and lights are arranged slightly different from Figure 5-1.

The system design minimizes the need for operator use of the console. A key switch gives added security from unauthorized use of the system by disabling the console. Thus, only authorized persons can enable the console by using the key switch.

The main functions of the operator console are to turn power on and off, reset the system, load the initial program information, and store or display information in storage or registers. Additional facilities exist for program debugging and hardware testing.

CONSOLE DISPLAY AREA

Indicator lights in the console display area (see Figure 5-2) may appear to be on during system operation. They are actually turning on and off at a rapid rate as a result of the internal speed of the processor. Therefore, data represented by indicators (except the test light) in the console display area is valid only when the processor is in the stop state.

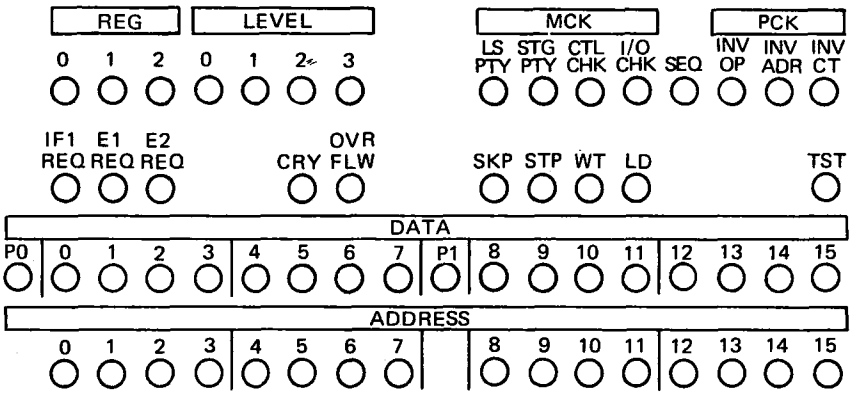


Figure 5-2. Console display area

REG Lights

The reg lights indicate, in binary notation, the index register being used by the current instruction. If all reg lights are off, the instruction either does not require an index register or is using the accumulator or instruction address register, depending on the specific instruction.

LEVEL Lights

The level lights indicate which one of the four interruption levels (0, 1, 2, or 3) is currently active.

Machine Check (MCK) Indicator Lights

LS PTY (Local Store Parity) Light

A parity error detected in one of the index registers turns on this indicator. A machine check interruption occurs if the check control switch is in the process position.

STG PTY (Storage Parity) Light

A parity error detected when data is read from storage turns on this indicator. A machine check interruption occurs if the check control switch is in the process position. (See 'Parity' in Chapter 2, "IBM 5010 Processor Module.")

CTL CHK (Control Check) Light

This light turns on under any one of the following conditions:

1. The processor has requested or taken more than one type of storage cycle simultaneously.
2. More than one interruption level is active simultaneously.
3. No interruption level is active, but the processor is requesting or taking storage cycles.

A machine check interruption occurs if the check control switch is in the process position.

I/O CHK (I/O Check) Light

A machine error condition that prevents further communication with the I/O modules turns on this light and causes a machine check interruption. (Refer to "SEQ (Sequence) Light" following.)

SEQ (Sequence) Light

This light operates in conjunction with the I/O check light. The sequence light is turned on if the machine error occurred during an interruption; the light is not turned on if the machine error occurred during an execute I/O instruction.

Program Check (PCK) Indicator Lights

INV OP (Invalid Operation) Light

This light turns on when an instruction contains an invalid value in the operation code, modifier, or function field. This results in a program check interruption. The MSP/7 Macro Library/Relocatable, in saving registers across calls to system routines, uses the program check interruption. This causes the INV OP light to flicker. See *MSP/7 Messages, Codes and Operating Procedures,* Order Number GC34-0023, for proper use of STOP ON ERROR in this environment. A system message is generated when an operation code violation occurs.

INV ADR (Invalid Address) Light

When the storage address register attempts to access a main-storage address that exceeds the limits of the storage installed in the system, this light turns on and a program check interruption occurs.

INV CT (Invalid Count) Light

This light turns on when an invalid count field is detected in a shift instruction, causing a program check interruption.

Cycle Indicator Lights

IF1 REQ (Instruction Fetch Request) Light

This light turns on to indicate that an instruction is about to be taken from storage (that is, the machine will next be in the instruction fetch cycle).

E1 REQ and E2 REQ (Execution Request) Lights

These lights turn on to indicate that the machine will next be in the first or second execution cycle, respectively.

Miscellaneous Indicators

CRY (Carry) Light

This light turns on when the carry indicator for the active interruption level is on.

OVRFLW (Overflow) Light

This light turns on when the overflow indicator for the active interruption level is on.

SKP (Skip) Light

This light turns on when conditions required for a successful skip by a skip or branch instruction are satisfied.

STP (Stop) Light

This light turns on when the machine stops at the end of its current cycle and enters the stop state under any one of the following conditions:

1. The stop key is pressed.
2. An error occurs when the check control switch is in the stop-on-error position.
3. The rate control switch is in the instruction-step position, or the stop-on-storage-address position, and the conditions applying to that position are met. (Refer to "Rate Control Switch" later in this chapter.)
4. The PSTP (stop) instruction is executed while the console controls are enabled. (See "Console Controls Switch" later in this chapter.)

Depressing the reset key on the operator console will also turn on the stop light as well as the wait light, indicating that the processor is in the manual wait state. (See 'Processor States' in Chapter 2, "IBM 5010 Processor Module.")

WT (Wait) Light

This light turns on when the machine is in either the wait or manual wait state, which can be caused by either of the following:

1. A level exit instruction is executed and no interruptions are pending at other levels (wait state).
2. The system is reset. In this case, the stop light is also turned on (manual wait state).

(See 'Processor States' in Chapter 2, "IBM 5010 Processor Module.")

LD (Load) Light

This light turns on when the processor enters the load state to perform an initial program load (IPL) operation. The load light remains on for the duration of the IPL operation, which can result from pressing the program load key or from the auto restart function.

The load light also turns on when an initial program load operation is performed from a host system via the 1130 host attachment, the Asynchronous Communications Control Attachment, or the Binary Synchronous Communications Adapter.

TST (Test) Light

This light turns on when the console controls are enabled and either the check control switch or the rate control switch is not set to the process position.

DATA Lights

These 18 lights consist of 16 data lights and the two associated parity lights (P0 and P1). The lights display data at a point in time determined by the setting of the check control and rate control switches. The displayed data is selected by the store/display switches.

ADDRESS Lights

These 16 lights display the last storage address accessed for data or for an instruction.

ROTARY SWITCH AREA

The rotary switch area is shown in Figure 5-3.

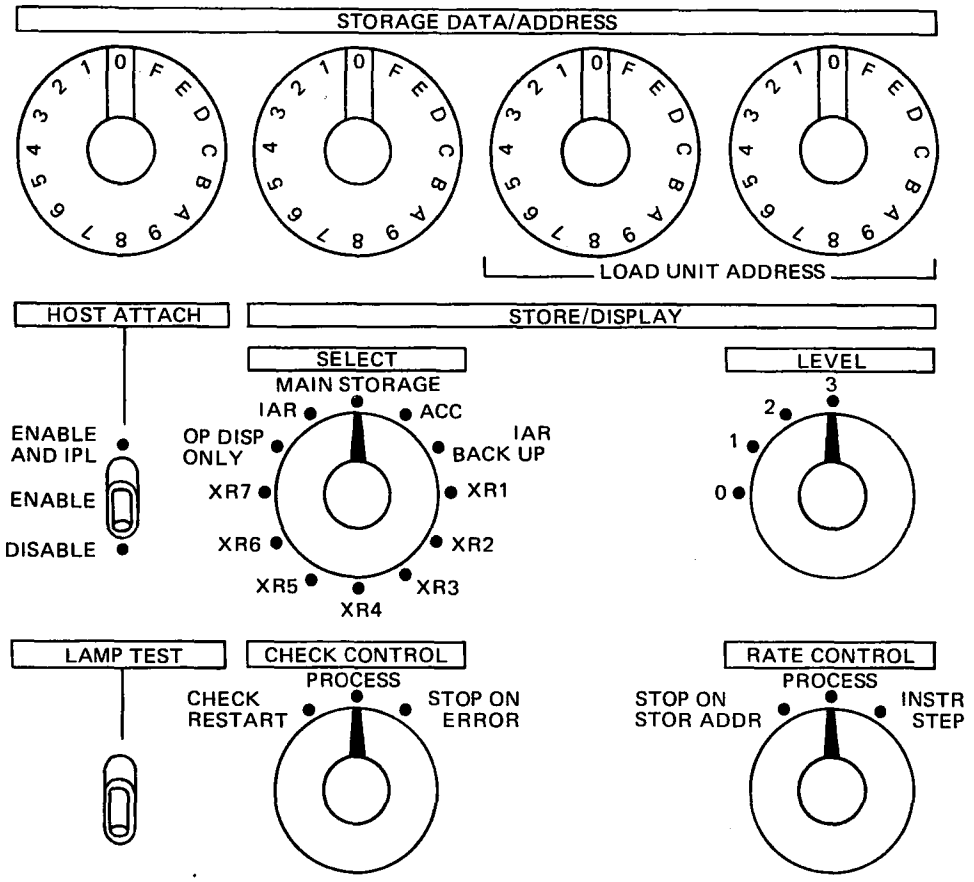


Figure 5-3. Rotary switch area

STORAGE DATA/ADDRESS Switches

These four rotary switches set up data for entry into the system via the operator console, or establish an address for use with the rate control switch. Each rotary switch has settings from 0 to F so that the data word or address to be established can be selected by its hexadecimal value.

Data set up by the rotary switches is stored into a storage location or the register indicated by the store/display select and store/display level switches. Storing occurs when the store key is pressed.

The two right-hand storage data/address switches are also labeled load unit address. These switches are used during any IPL operation to select the source of the storage load. The load unit address switches must be set to 00 for an IPL from either the operator station tape reader or a host processor. For an IPL from a disk storage module, the load unit address switches must be set to the I/O module address containing the disk storage module to be used. (See "PROGRAM LOAD Key" later in this chapter.)

HOST ATTACH Switch

This switch controls the operation of the host processor connection (Asynchronous Communications Control Attachment or Binary Synchronous Communications Adapter or 1130 host attachment). See Chapter 7, "Asynchronous Communications Control Attachment," or Chapter 8, "Binary Synchronous Communications Adapter," or Chapter 16, "1130 Host Attachment" for more detail on the functions controlled by the three positions of this switch.

STORE/DISPLAY SELECT Switch

This 12-position rotary switch governs which storage location or register is addressed from the console for storing and/or displaying information. Information contained in the selected location or register is displayed by the console data lights. Information to be stored is set up in the storage data/address switches. Storing occurs when the store key is pressed. A register is automatically displayed by the data lights when the machine stops, depending on the setting of the store/display select and store/display level switches. A storage location can be displayed by the data lights by pressing the main storage display key when the machine is in the stop state.

The storage locations or eleven registers that can be selected by the store/display select switch are:

1. OP (operation register). This register contains the first word of the last instruction executed. The op register can be selected only for the display of information.
2. IAR (instruction address register). This register is the main incrementing IAR, not the back-up IAR associated with priority levels 1 to 3.
3. MAIN STORAGE (main storage location). This location is specified by the IAR. The contents of the specified storage location are displayed when the main storage display key is pressed. To store information, the storage location must first be set into the IAR.
4. ACC (accumulator). This position selects the accumulator associated with levels 0 to 3 as determined by the store/display level switch.
5. IAR BACK UP. This position selects the backup instruction address register (IARB1 to IARB3) associated with levels 1 to 3 as determined by the store/display level switch. A backup IAR can be selected only for the display of information.

6.–12. XR1 to XR7. These positions select index registers 1 to 7 associated with levels 0 to 3 as determined by the store/display level switch.

STORE/DISPLAY LEVEL Switch

This four-position switch is marked from 0 to 3 so that the appropriate priority level can be selected. The level switch is used in conjunction with the select switch. Each priority level has an accumulator and a set of seven index registers. Priority levels 1 to 3 also have a backup IAR. The level switch determines what priority level is selected for the store/display switch setting.

LAMP TEST Switch

This switch tests the console indicator lights. All indicator lights in the console display area should remain on as long as the switch is held in the on position.

CHECK CONTROL Switch

This three-position switch has the following settings:

1. PROCESS. Normal switch position; a program check, machine check, or power/thermal warning condition causes an interruption.
2. STOP ON ERROR. A program check or machine check error causes the system to stop at the end of the machine cycle in which the error is detected. The stop light is then lit and the error condition displayed by the appropriate console indicators. A system reset must occur in order to clear this error.
3. CHECK RESTART. A machine check or program check error causes system reset and the restart of program execution on priority level 3 at address 0000. This provides a looping facility for the customer engineer.

Note: If the console controls are disabled, the test light does not turn on and the stop-on-error and check-restart switch positions function the same as the process position.

RATE CONTROL Switch

The three positions of this switch have the following effects:

1. PROCESS. Normal position; permits continuous program execution.
2. STOP ON STOR ADDR (stop on storage address). Whenever the contents of the processor storage address register match the address designated by the storage data/address rotary switches, the machine stops at the end of the current instruction and the stop light is turned on. If the storage data/address switches designate the address of an operand, the address displayed by the address lights will be one greater than the address designated in the switches.
3. INSTR STEP (instruction step). Operation of the start key causes one instruction to be executed. The machine then stops and the stop light turns on.

Note: If the console controls are disabled, the test light does not turn on and the stop-on-storage-address and instruction-step switch positions function the same as the process position.

CONSOLE FUNCTION SWITCHES AND INDICATORS

The console function switches and indicators are shown in Figure 5-4.

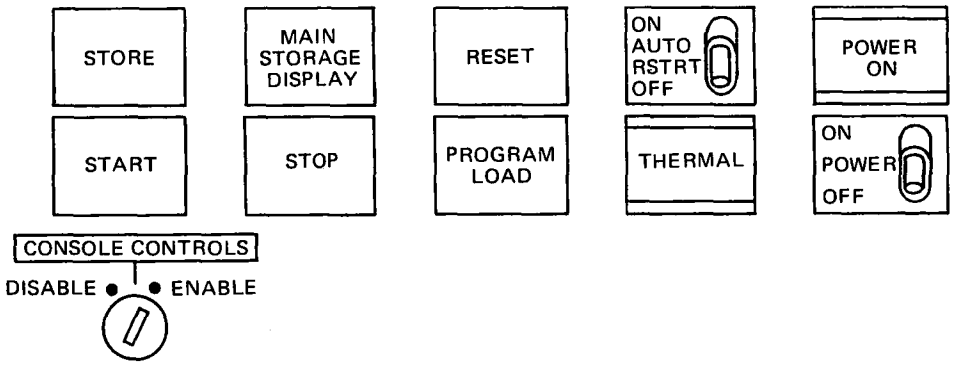


Figure 5-4. Console function switches and indicators

STORE Key

Pressing this key stores the data represented by the storage data/address rotary switches into the register or storage location specified by the settings of the store/display select and store/display level switches.

If main storage is selected, the location to receive data must first be set into the IAR by using the storage data/address rotary switches. Sequential depression of the store key automatically increments the address if main storage is selected. The store key operates only when the machine is in the stop state.

MAIN STORAGE DISPLAY Key

Pressing this key causes the data lights to display the data located at the main storage address specified by the IAR. Sequential depression of this key automatically increments the address.

Register contents are automatically displayed by the data lights when the machine is stopped. No key need be pressed. The register that is displayed is determined by the settings of the select and level switches.

The main storage display key operates only when the machine is in the stop state and the store/display select switch is in the main storage position.

RESET Key

The following actions occur when the reset key is pressed:

1. The entire System/7 is reset, with the exception of digital and analog output points.
2. The System/7 processor enters the manual wait state and all error and interruption conditions are cleared (set to 0). Internal registers (accumulators, index registers, and backup instruction address registers) are not reset.
3. The priority interrupt mask is set so that all interruptions are enabled.

AUTO RSTRT (Automatic Restart) Switch

This is a two-position toggle switch. When the switch is set in the off position, operator intervention is necessary to restart the power system after a shutdown due to loss of line voltage.

If the switch is set to the on position, the enclosure will attempt to restart the power system for an indefinite period starting one second after an initial power failure. When power is restored in the enclosure, the modules are reset automatically. If the enclosure contains a processor module, the processor attempts to perform an IPL operation. If the power failure and automatic restart occurred in an expansion enclosure, no IPL operation is performed. At the end of the IPL function, bit 0 of the level 3 accumulator will be set on to indicate that the IPL was started by the auto restart function, not the program load key.

Operating note. If an automatic IPL is not concluded, press the RESET key before pressing the PROGRAM LOAD key.

POWER ON Light

The power on indicator is turned on when power is in the system.

START Key

Pressing the start key causes processing to resume on the same level that was active at the time the machine was stopped. The amount of processing accomplished depends upon the setting of the rate control switch. When the machine is in the manual wait state (both the stop and the wait lights are on), pressing the start key causes the processor to activate level 3 and begin program execution at the storage location specified by the IAR. Note that if the processor was in the wait or manual wait state and the machine was stopped by depressing the stop key, depressing the start key causes the processor to return to the wait or manual wait state.

STOP Key

Pressing the stop key stops the machine and turns on the stop light after completion of the current instruction.

PROGRAM LOAD (IPL) Key

Pressing this key resets the system as if the reset * key were depressed, and places the processor in the load state. The load light turns on and remains on for the duration of the IPL operation. The IPL data source must be specified by the load unit address switches (the two right-hand storage data/address switches) before the program load key is pressed.

Data is transferred from the data source and stored in main storage starting at location 0000. Upon completion of the IPL operation, the load light turns off, interruption level 3 is activated, and instructions are executed beginning at location 0000. When instruction execution begins, bits 10 to 15 of the level 3 accumulator will contain the load unit address, and bit 0 will be set off to indicate that the IPL operation was started by pressing the IPL key, not by the auto restart function.

Note: The program load key is inoperative if the host attachment switch is in the enable and IPL position.

THERMAL Light

The thermal light indicates an overtemperature or undertemperature condition in the enclosure.

POWER Switch

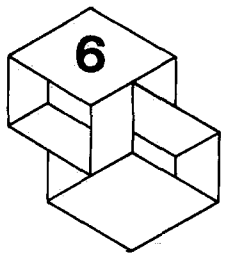
Placing this toggle switch in the on position applies power to the power system and turns on the power-on light. In the off position, line power is removed and the power-on light is turned off.

CONSOLE CONTROLS Switch

When this switch is in the disable position, all console functions are disabled except the power, host attach, and auto restart switches. A key must be inserted to turn the switch to the enable position. The key cannot be removed unless the switch is in the disable position.

* See operating note under AUTO RSTRT switch description.

CHAPTER CONTENTS



The processor module contains two 16-bit interval timers. Each timer is a one-word binary counter that decrements the value of its contents once each 50-microsecond interval. Upon decrementing from 0 to −1, the timer presents an interruption request to the processor. The interruption request occurs when the timer has counted *one more* count than the specified number. Unless the timer is stopped, it continues to count and presents an interruption request again whenever the transition from 0 to −1 is made. The timer value follows the sequence −1, −2, . . . , −32768, +32767, . . . , +1, 0, −1, . . .

Though timers are automatically decremented by the machine, they are separately controlled by programming. Timers can be started, stopped, read, or set to a value by I/O commands. A timer that is running can be read without disturbing its operation, but a timer must be stopped before it can be set to a new value.

The interval timers are reset to −1 (hexadecimal FFFF) when System/7 power is turned on. If a value is not set in a timer after system power is turned on and the timer is started, the timer will count the maximum time value (see sequence above) before presenting an interruption request to the processor.

The timers are incremented by a crystal controlled oscillator. The accuracy of this oscillator is as follows:

Nominal Accuracy

10 mHz +0.0%–0.025%
(Maximum loss of 21.6 seconds in 24 hours.)

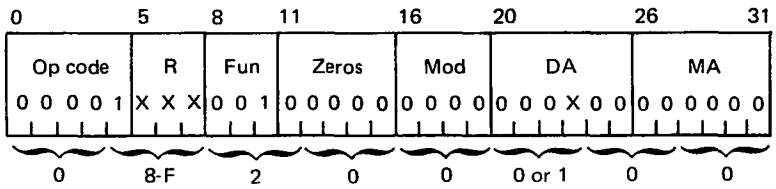
Calibration Accuracy

9,999,055 Hz ±0.003% with a 90% confidence factor

I/O COMMANDS

Both interval timers are prepared simultaneously by a single prepare I/O command. The prepare command assigns the same priority level and sublevel to both timers, with a module address of 000000 and a device address of either 000000 or 000100 for timer 0 or 1 respectively. The prepare command is described further in Chapter 4 under "Input/Output Instruction." There is no interrupt status word associated with the interval timers because an interrupt from an interval timer has only one meaning.

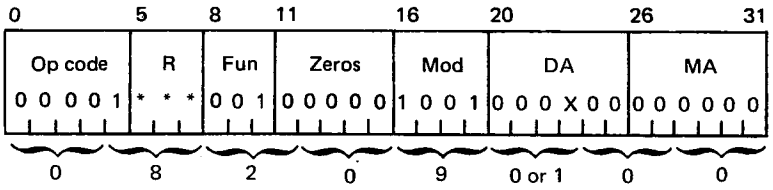
Set Timer



The addressed timer (specified by the device address field) is loaded with the value residing in the index register (R), or the accumulator if R=000. This value is the number of 50-microsecond intervals to be counted, after a start timer command is issued, before the timer interrupts the system. Timers must be stopped before they are set to a new value.

Condition code 2 is set if the addressed timer is already running or has an interruption pending.

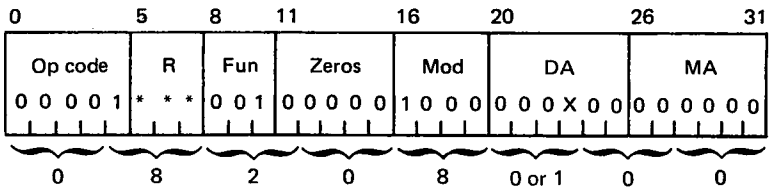
Start Timer



The addressed timer (specified by the device address field) is started. The R field is ignored. When the timer completes counting the specified number of 50-microsecond timer intervals, an interruption request is presented to the processor.

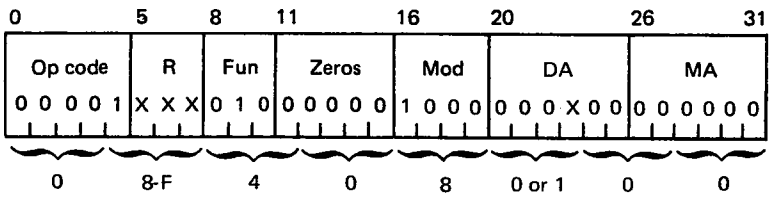
Condition code 2 is set when the start timer command is issued to a timer that has an interruption pending, or is already on and counting.

Stop Timer



The addressed timer (specified by the device address field) is stopped. The R field is ignored. Timers must be stopped before they are set to a new value.

Read Timer



The count value in the addressed timer (specified by the device address field) is stored into the index register (R), or the accumulator if R=000. No interruption occurs and the timer continues to count.

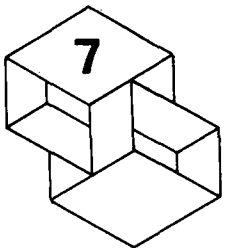
TIMER INTERRUPTIONS

Timer interruption requests use the same interruption mechanism as the I/O devices. A read-ISW command to the timers always records an interrupt status word that contains all 0's because no errors can occur during timer operations and timer interruptions have only one meaning.

When the timers request an interruption, the contents of the device address field in the interruption identification will correspond to the proper timer (0 or 1), even though both timers were prepared by a single prepare I/O command.

In the event of simultaneous interruptions, timer 0 has priority over timer 1.

CHAPTER CONTENTS



The Asynchronous Communications Control Attachment (ACCA) is an optional feature that resides in the IBM 5010 Processor Module Models A and E.

The ACCA controls the transfer of serial data to and from a host processor at a remote location via a data set and communications line facility. The ACCA can be used for connecting a System/7 to telecommunication equipment or other processors having compatible adapters. Any number of System/7 configurations (each with an ACCA) can function as satellite processors to a System/360 (model 25 and larger), a System/370, or an 1800 Data Acquisition and Control System. (For the user who is not familiar with data communications, the manual *Teleprocessing System Summary*, Order No. GA24-3090, may be helpful.)

Data transmission is serial-by-bit, using the start/stop method of character and bit synchronization. The code used is the IBM Paper Tape Transmission Code/Extended Binary Coded Decimal (PTTC/EBCD). (See Appendix D.) Figure 7-1 shows the data word and PTTC/EBCD bit assignments within the second byte of the word. Odd parity must be used for bits 9 to 15. Therefore, on transmit, the character parity bit (C-bit) must be provided by the program in the System/7 processor; on receive, the ACCA checks for correct parity.

Data rates, selected by machine pluggable options, are 14.8 or 66.7 characters per second. These character rates correspond to bit rates of 134.5 and 600 bits per second respectively.

The ACCA operates in half-duplex mode; that is, the System/7 or the host processor can either transmit or receive data, but cannot transmit and receive simultaneously.

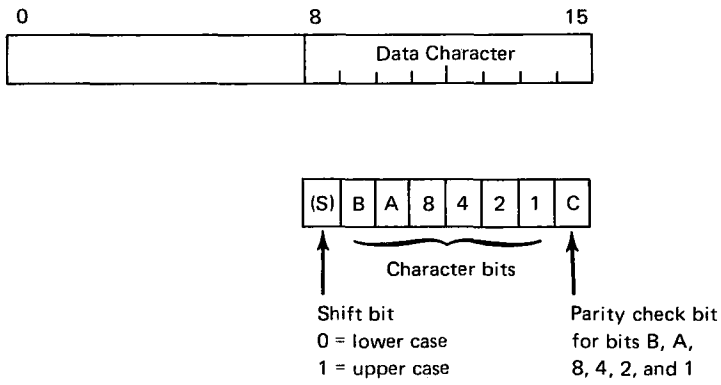


Figure 7-1. ACCA data word

IBM COMMUNICATION LINE ADAPTERS

The ACCA can communicate with a host processor over private lines, leased common-carrier facilities, or switched voice-grade common-carrier lines.

The various line adapters that can be installed in the ACCA are shown below.

IBM Line Adapters

Type	*Rate (bits/sec)*	*Line Type*	*Maximum Line Length*
2B limited distance	600 maximum	2-wire	8.25 miles
1A leased line	600 maximum	2-wire	No limit
1B leased line	600 maximum	4-wire	No limit

IBM World Trade Modems

Type	*Rate (bits/sec)*	*Line Type*	*Type of Service*
IBM 3976-I	200 maximum	2- or 4-wire	Private or leased
IBM 3976-II	200 maximum	2-wire	Switched
IBM 3976-III	1200 maximum*	2- or 4-wire	Private or leased

*System/7 supports 600 bps maximum

In conformance with Electronic Industries Association (EIA) Standard RS232C within the United States, and Consultive Committee on International Telephone and Telegraph (CCITT) V24 Standards outside the United States, an attachment option provides the following eight interface signals for non-IBM modems:

1. Transmitted data
2. Received data
3. Request to send
4. Clear to send
5. Data terminal ready
6. Data set ready
7. Protective ground
8. Signal ground

Some modems disconnect when the data terminal ready signal is deactivated in the ACCA. This signal can be deactivated for 1 second by a prepare I/O command that prevents interruption requests, provided interruption requests were already permitted by a prior prepare I/O command. At the end of the 1-second interval, the data terminal ready signal again becomes active, but interruption requests are permitted again only if another prepare I/O command has done so. During the one-second time period, the System/7 is unavailable to any processor that may be attempting to communicate with it. The data terminal ready signal is not affected by a system reset.

A precise exchange of control characters between computers is necessary to establish and maintain teleprocessing communications. This exchange of control characters is called line control. One of its functions is to prevent two or more stations, terminals, or System/7s from attempting to use the line simultaneously (line contention).

Line Control Characters

As the following table shows, the line-control characters (signals) are represented in shorthand form as acronyms. Some of the acronyms also have equivalent encircled letters. These are referred to as "circle C, circle D," etc. This form is used in programming, as well as in communications and line-control discussions.

Character Function	*Acronym*	*Binary*	*Hexadecimal (PTTC)*
End of transmission	EOT Ⓒ	0001 1111	1F
End of address	EOA Ⓓ	0001 0110	16
End of block	EOB Ⓑ	0011 1101	3D
Positive response	YAK Ⓨ	0111 0110	76
Negative response	NAK Ⓝ	0100 0000	40
Start of address	SOA Ⓢ	0011 0111	37
Upshift	US	0001 1100	1C
Delete	DEL	0111 1111	7F
Downshift	DS	0111 1100	7C
Space	SP	0000 0001	01

All characters except US and DS are presented as data when the ACCA is in standby mode. EOB, the first EOT at the end of transmission, US, YAK, NAK, SOA, and SP (SP when used in addressing and polling sequences) are not presented as data.

The function of line control character signals is as follows:

Acronym	*Function*
EOT Ⓒ	Indicates end of transmission; resets ACCA to standby mode and, if station control option is installed, resets ACCA to unselected status.
EOA Ⓓ	Switches ACCA from receive mode to transmit mode and vice versa. This signal starts the LRC counter at both the sending and receiving terminals. The EOA signal is not included in the LRC check that follows it.
EOB Ⓑ	Indicates the end of a unit-block of text. This is followed by the LRC character, which provides an LRC check comparison at the receiving system with the EOB character included in the check.
YAK Ⓨ	Indicates a positive response (Yes). It is coded as a period character and indicates a positive response to an address or a positive answer to an EOB/LRC sequence.

NAK	(N)	Indicates a negative response (No). It is coded as a hyphen character and indicates a negative response to an address, a poll, or an EOB/LRC sequence.
SOA	(S)	Indicates a start-of-address condition. It is coded as a comma character and is used when the station control feature is installed.
US		Indicates that the characters that follow will be uppercase until a DS, EOB, or EOT character is decoded.
DEL		One of the line control characters used in a sequence to cause System/7 to enter IPL mode from standby mode. This sequence is as follows: 1. EOA 2. US 3. DEL 4. DS
DS		Indicates that the characters that follow will be lowercase until a US is decoded. Two consecutive DS characters put the ACCA into binary mode.
SP		This signal is coded as a character with no bits. It is used in station control addressing and polling.

The ACCA uses the line control characters in a manner similar to the way in which the IBM 2740 Communication Terminal Model 1 uses them. For more detail on line control of the 2740 Communication Terminal, refer to the manual *IBM 2740 Communication Terminal Models 1 and 2 Component Description,* Order No. GA24-3403.

STATION CONTROL

A station control option installed in the ACCA connects the System/7 to a communications line that is shared with other teleprocessing devices or systems. The System/7 is selected by a host processor when the host transmits the sequence of characters EOT, SOA, ADDRESS (unique for each device on the communications line), and SPACE (see Figure 7-2).

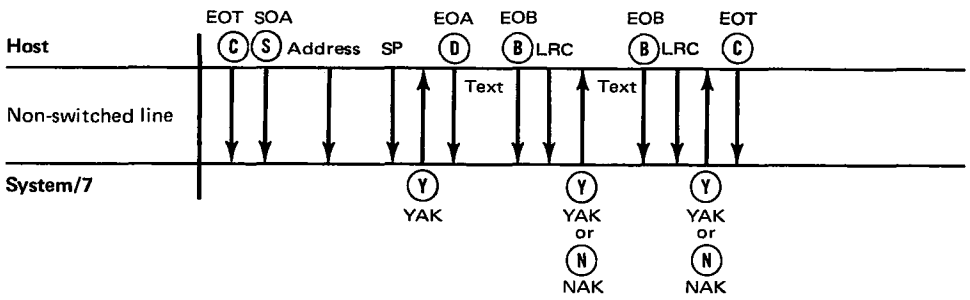


Figure 7-2. Station control (addressing) operation

The ACCA must be selected in order to present received data to the System/7 program or to transmit data to the host processor. If the ACCA receives or transmits an EOT character, the ACCA is reset to an unselected condition. When the ACCA recognizes its address in the addressing sequence, it transmits a positive or negative acknowledgement depending on the position of the host attach switch on the operator console and whether interruption requests are permitted. If the host attach switch is set in the enable and IPL position, the ACCA transmits a positive acknowledgement. If the switch is in the enable position and interruption requests are permitted, the ACCA transmits a positive acknowledgement. If the switch is in the enable position and interruption requests are prevented, the ACCA transmits a negative acknowledgement to the host processor. The ACCA does not send an acknowledgement if the host attach switch is set in the disable position. The EOT character in the sequence mentioned previously is not required if the host processor addresses another device and receives a negative acknowledgement prior to addressing the System/7.

The ACCA is reset to an unselected condition if the first character received from the host (after the addressing sequence) is not an EOA character.

System/7 can transmit to the host processor only when the ACCA has been polled by the host processor. The first polling sequence of characters from the host processor must be EOT, ADDRESS, and SPACE (see Figure 7-3). If the first polling sequence is acknowledged with a negative response, subsequent polling sequences can be ADDRESS and SPACE until a positive acknowledgement is received by the host.

When the ACCA recognizes its address in the polling sequence, it presents an interruption request to the processor with ISW bit 0 set. The System/7 program would normally start a transmit sequence of EOA, text, EOB as a result of this interruption request. If the first character in the message transmitted is not EOA, the ACCA is reset to an unselected condition and no more interruption requests are presented to the processor. Message ending sequences are the same as sequences for the ACCA without the station control option.

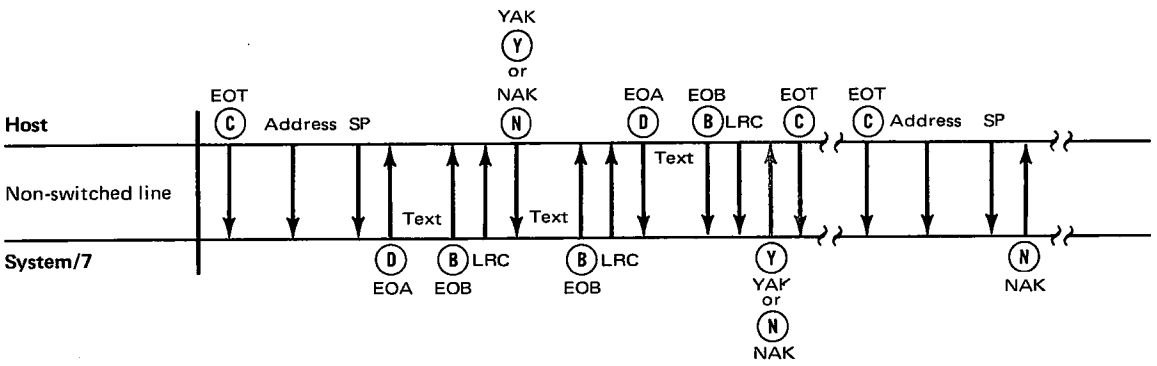


Figure 7-3. Station control (polling) operation

I/O COMMANDS

The ACCA is programmed using the prepare I/O command as described in Chapter 4 under "Input/Output Instruction." Data is transferred between the processor and the ACCA by immediate read and immediate write commands. The setting of the modifier field bits in an immediate command further defines the operation to be performed.

All I/O commands to the ACCA must have a device address of 001100 and a module address of 000000.

The terms "status" and "mode" as used in the I/O command descriptions have different meanings. While in either transmit or receive mode, the ACCA accumulates a logitudinal redundancy check (LRC) character. Figure 7-4 shows a line control sequence for both point-to-point and dial-up operation. These methods are the same but in dial-up operation a communication line is connected and disconnected through a data set.

The ACCA enters transmit status upon receiving a transmit control command in order to establish a data link and to block any possible incoming data. The ACCA enters transmit mode when it detects an EOA character in a transmit character command. This EOA character is not accumulated in the LRC register, but any following EOA characters are considered as data and accumulated in the LRC register.

The ACCA enters receive status when it detects a start bit from a remote processor. The ACCA returns to standby mode after the received character is deserialized unless that character is an EOA. If the character is an EOA, the ACCA enters receive mode. This EOA character is not accumulated in the LRC register, but any following EOA characters received are considered as data and accumulated in the LRC register.

The ACCA is in standby mode when it can accept transmit commands or receive data. When in standby mode, the ACCA decodes all characters received except upshift and downshift (including line noise) and presents the decoded characters to the program as data.

The ACCA exits from transmit or receive mode and then enters standby mode under any one of these conditions:

- The ACCA detects an end-of-transmission (EOT) character
- A time-out interruption request occurs
- A reset ACCA command is issued by the System/7 program

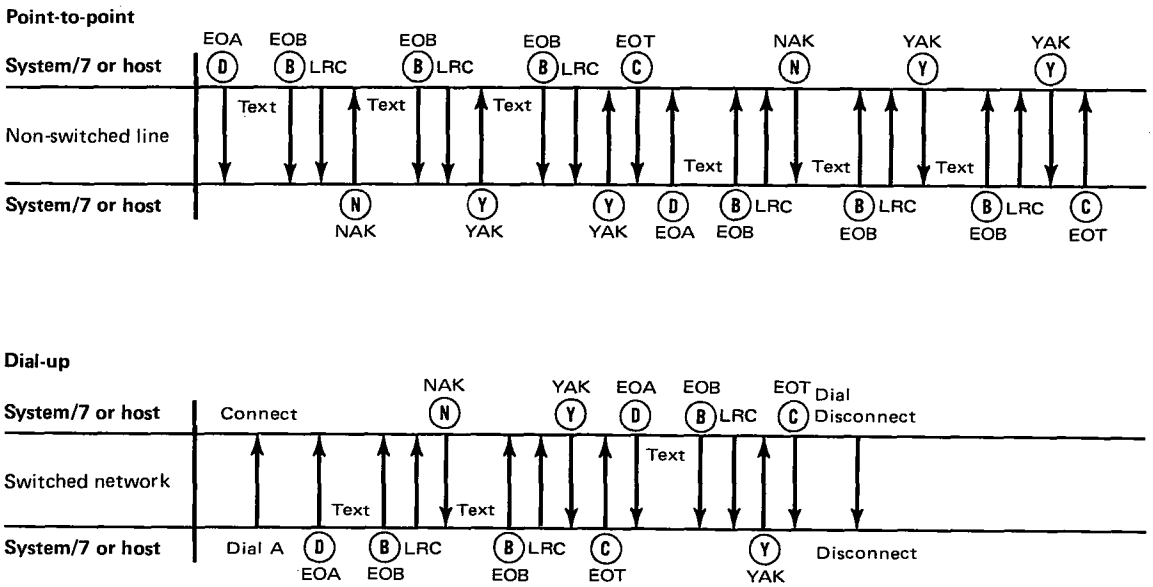


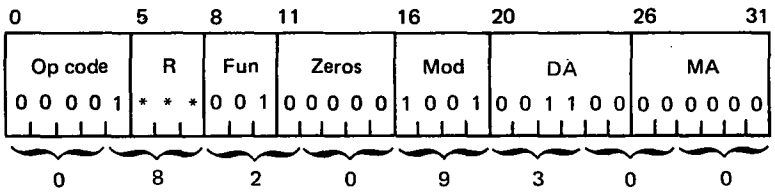
Figure 7-4. Point-to-point and dial-up operation

Data and control characters are transmitted one byte at a time. On output, bits 8 to 15 of the index register or accumulator are sent to the ACCA and bits 9 to 15 are transmitted over the communications line. Register bit 8 indicates to the ACCA if the character is uppercase or lowercase. Bit 8 is set to 1 if the character is uppercase; bit 8 is set to 0 if it is lowercase. The System/7 program does not need to insert an upshift or downshift character. The ACCA transmits an upshift character if bit 8=1 or a downshift character if bit 8=0 prior to transmitting the character contained in bits 9 to 15 of the register.

On input, received data replaces the contents of bits 9 to 15 of the index register or accumulator. The ACCA sets bit 8 on when an upshift character is received. This bit signifies uppercase and will remain on for all subsequent characters until a DS, EOB, or EOT character is received. Bits 0 to 7 are set to 0 if the ACCA is not operating in binary data mode. (Binary data mode is explained later in this chapter.) If the ACCA *is* operating in binary data mode, received data replaces the contents of bits 0 to 15 of the index register or accumulator.

Interruption requests are made to the System/7 processor only if the host attachment switch on the System/7 console is in the enable or enable and IPL position. If the switch is in the disable position, condition code 3 is returned to the processor whenever the ACCA is addressed.

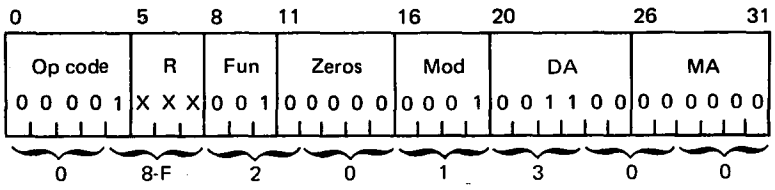
Transmit Control



This command puts the ACCA into transmit status. The R field is ignored. If the ACCA is in receive status or receive mode, it returns condition code 1 to the processor and sets bit 4 (ACCA check) in the direct control channel status word.

The ACCA attempts to establish a data link with the remote processor, during which time the ACCA remains busy. A time limit of 16.5 seconds is allowed to establish the data link and to issue a transmit character command. If the data link is established, the ACCA presents an interruption request after setting the ready-to-transmit bit (bit 6) in the ISW. If no link is established, a time-out interruption request (ISW bit 5 is set) is presented, instead of the ready-to-transmit interruption request.

Transmit Character

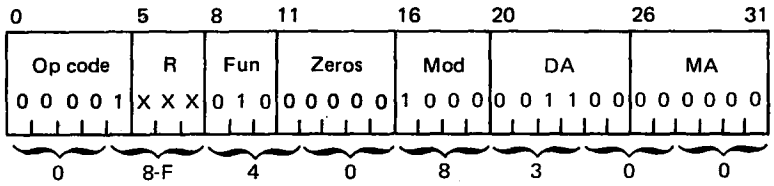


This command transfers bits 8 to 15 of the index register (R), or the accumulator if R=000, to the ACCA. If the ACCA is in receive status or receive mode, it returns condition code 1 to the processor and sets bit 4 (ACCA check) in the direct control channel status word. When character transmission begins, the ACCA presents an interruption request after setting the ready-to-transmit bit (bit 6) in the ISW.

For a multiple-character transmission, a single transmit control command should precede the transmit character command(s) in order to establish the data link and place the ACCA in transmit status. The first EOA character transmitted places the ACCA in transmit mode.

For a single-character transmission, the transmit character command can be used without a preceding transmit control command. In this case, however, only a device-end interruption request occurs (ISW bit 13 set), and no ready-to-transmit interruption request results from the single transmit character command. After a single-character message is transmitted in this manner, the ACCA returns to standby mode.

Read Character

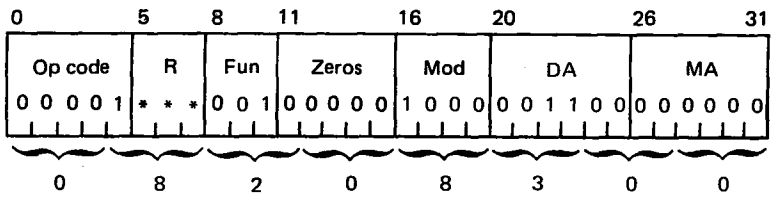


This command stores the character received from the host processor into bits 8 to 15 of the index register (R), or the accumulator if R=000. If the ACCA is in transmit mode, it returns condition code 1 to the processor and sets bit 4 (ACCA check) in the direct control channel status word.

Bits 0 to 7 of the register or accumulator contain 0 if the ACCA is not operating in binary data mode. If the ACCA *is* operating in binary data mode, the read character command stores a 16-bit data word in the index register or accumulator.

The read character command is normally issued following a character-received interruption request from the ACCA.

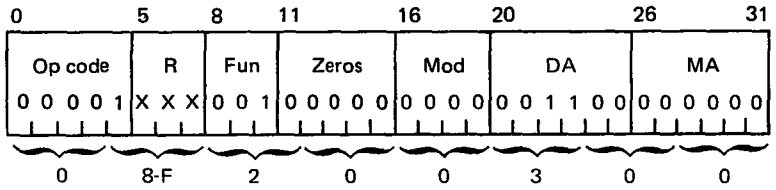
Reset ACCA



This command performs a halt I/O function in the ACCA. Any pending interruption, device status, and all other controls are reset, with the single exception of data sent by a prepare I/O command. The R field in the command is ignored.

Following this reset command, the ACCA is in the standby mode of operation if interruptions are allowed, and the host attachment switch is in the enable or enable and IPL position. Incoming messages can be received, or transmit operations can begin, under control of a System/7 program. If interruptions are prevented, an IPL sequence will still be recognized when the ACCA is reset to standby mode. (See "Initial Program Load (IPL)" in this chapter.)

Diagnostic Write

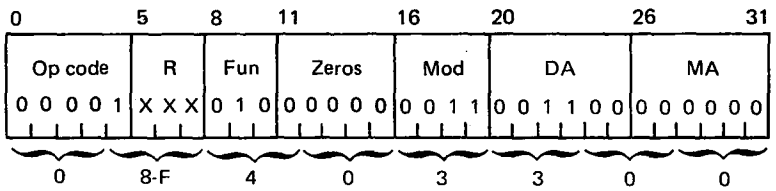


This command supplies bit 8 of the index register, or accumulator if R=000, to the output of the transmit circuits. The output must be manually connected to the ACCA input for testing of the receive circuits.

Nine diagnostic write commands are needed to create a character in the ACCA before an interruption request can be presented to the processor. Bit 8 of the index register or accumulator determines the setting of the bit transmitted by the ACCA. If bit 15 of the index register or accumulator is set to 1, the ACCA exits from diagnostic mode. Each of the nine diagnostic write commands represents one of the nine bit-times required to transmit a character—start, 7 character bits, and stop. The following chart gives the settings of bits 8 and 15 for transmitting a character with diagnostic write commands.

Command No.	*Name*	*R-Reg Contents*
1	Start	Bit 8=0; bit 15=0
2–8	Character bits	Bit 8=x; bit 15=0
9	Stop	Bit 8=1; bit 15=0
Last	Exit diagnostic mode	Bit 8=x; bit 15=1

Read ISW



This command stores the 16-bit interrupt status word (ISW) associated with the ACCA into the index register (R), or the accumulator if R=000.

Interrupt Status Word (ISW): Bits in the interrupt status word are set on to indicate the operating status of the ACCA and to notify the system of errors detected. Interruptions result from setting on any ISW bits except 1, 8, 10, and 12. When the interruption request is serviced by the program, subsequent execution of any command to the ACCA resets the ISW. The ISW is also reset by system reset.

Descriptions of significant bits in the ACCA ISW follow. An asterisk (*) next to the bit number indicates that the summary status indicator is also set when the resulting interruption request is presented to the processor. If the asterisk is enclosed in parentheses, the summary status indicator is set only when a specific condition sets the ISW bit.

Significant Bits	*Meaning*
0*	Attention. The ACCA has recognized a poll from the host processor. This bit can be active only when the station control option is installed and the ACCA recognizes its own address. ISW bit 13 is also set along with this status bit.
1	Serial transmit data. This bit reflects the condition of the serial transmit line at all times. Bit 1 is used as a diagnostic aid for the transmit circuits of the ACCA and, consequently, has meaning only to a diagnostic program. This bit may or may not be set when the ISW for the ACCA is read. No interruption requests are presented when this bit is set.
2*	Overrun. A character was received before the previous character was stored by the System/7. ISW bit 13 is also set if the interruption request from the previous character is still pending.
3*	End of block. When in receive mode, the ACCA has recognized an end-of-block (EOB) control character followed by a longitudinal redundancy check (LRC) character. If the received LRC character does not compare with the LRC character generated by the ACCA, ISW bit 14 (data check) is also set.
4 (*)	Character received. One of the following conditions occurred: 1. If the ACCA is in receive mode, a character other than end of block (EOB), end of transmission (EOT), longitudinal redundancy check (LRC), upshift (US), or downshift (DS) was received. ISW bit 10 (receive mode) is also set when the resulting interruption request is presented to the processor. 2. If the ACCA is in receive status, a character other than US or DS was received. The summary status indicator and ISW bit 13 are also set when the resulting interruption request is presented to the processor. 3. If the ACCA is operating in binary data mode, a 16-bit binary word has been assembled by the ACCA. The summary status indicator and ISW bits 8, 10, and 13 are also set when the resulting interruption request is presented to the processor. *Note:* ISW bit 4 can be set when the ACCA has received line noise as data.
5*	Time out. One of the following does *not* occur within a 16.5-second time limit: 1. The ACCA responds to a transmit control command. 2. The System/7 issues a transmit character command when the ACCA is in transmit status or mode. 3. The ACCA receives a character when in receive mode. 4. The ACCA receives a response to the transmission of an EOB character.
6 (*)	Ready to transmit. Either of the following occurs: 1. The ACCA is ready to accept a new character for transmission. If the ACCA is not in transmit mode at this time, the summary status indicator is also set. 2. A transmit control command is issued and the ACCA senses that the data set is ready to send.

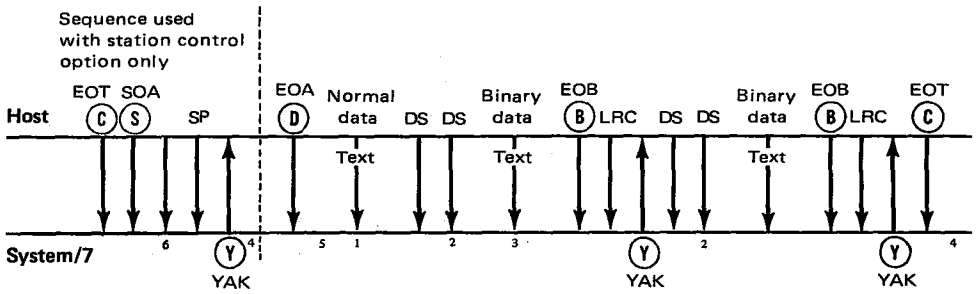
Significant Bits	*Meaning*
7*	Data set error. The data set has become inactive while the ACCA is in transmit or receive mode. ISW bit 13 is also set if the interruption request from the previous character is still pending. (ISW bit 7 is not used with IBM line adapters.)
8	Binary data mode. The ACCA has recognized two consecutive DS characters while in receive mode; subsequent data received will be in binary mode. This status bit is set along with other ISW bits as long as the ACCA is in binary data mode. This bit does not generate an interrupt.
10	Receive mode. The ACCA is in receive mode. This bit does not generate an interrupt.
12	Device busy. The ACCA is transmitting or receiving a character as a result of a transmit or a read character command. The status of this bit does not cause interruption requests nor does a read ISW command reset the bit.
13	Device end. This status bit is set on when all interruption requests are presented to the processor with the possible exceptions of overrun (bit 2) and data set error (bit 7). ISW bit 13 will be set when an overrun or data set error occurs only if the pending interruption request that set bit 13 has not been accepted by the processor. Bit 13 is the only ISW bit set when the ACCA detects an end-of-transmission (EOT) character and the ACCA is in receive mode. At this time the ACCA enters receive status and subsequent characters (including EOT) are presented to the program as described previously under condition 2. for ISW bit 4. The summary status indicator is also set if bit 13 is the only ISW bit set.
14*	Data check. A negative response, a synchronization error, a longitudinal redundancy check (LRC), or a vertical redundancy check (VRC) error has occurred. An LRC error is indicated by the presence of ISW bit 3 (end of block) along with the data check indicator. VRC and synchronization errors are indicated by the presence of ISW bit 4. A negative response is indicated by the presence of ISW bit 6.

BINARY DATA MODE

Binary data can be received by the ACCA if it is in receive mode and a special sequence of two consecutive downshift characters precedes the binary data received in the message (see Figure 7-5). Following this sequence, binary data is received from the host by transferring four binary bits from each character to create one hexadecimal character.

Character bit positions, B, A, 4, and 2 are the four bit positions used for a hexadecimal character. The ACCA assembles four hexadecimal characters, inserting the first character into bits 12 to 15, and transfers the resulting 4-character word (16 bits) to the System/7 processor storage. The fourth character assembled must have character bit position 1 set on to signal the ACCA that a 16-bit word is ready to be transferred to the processor. If a 1-bit is not detected, the ACCA continues converting characters to binary data, overlaying the previously assembled data word.

If a bit is received in character bit position 1 before four characters have been assembled, the ACCA stops assembling characters, fills the remaining high-order bits with 0's to complete the 16-bit word, and then presents an interruption request to the System/7. The host system and the ACCA communicate in a normal manner, observing all the rules for line control, error checking, and other character constraints. The ACCA exits from binary data mode when it receives an end-of-block (EOB) or end-of-transmission (EOT) character.



Notes:

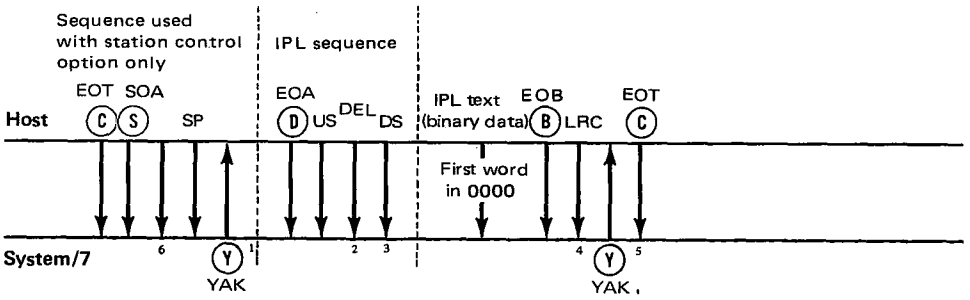
[1] Bits 8–15 of the index register or accumulator
[2] ACCA in binary data mode
[3] Bits 0–15 of the index register or accumulator
[4] ACCA in standby mode
[5] ACCA in receive mode
[6] Address

Figure 7-5. Binary data mode operation

The host processor determines when to initiate an IPL to the System/7 through the ACCA. The IPL is then initiated by means of a special IPL sequence of characters (see Figure 7-6). This IPL is allowed only when the host attach switch on the System/7 console is in the enable and IPL position, and the ACCA is in standby mode of operation. The System/7 program can, however, reject the IPL when the same special character sequence is received by the ACCA, as described later in this discussion.

An IPL is initiated by a host processor that sends the unique sequence of characters EOA, US, DEL, DS at the beginning of a message. The EOA character places the ACCA in receive mode and the US character is a control character to the ACCA. The DEL character is decoded as the first data character. If interruption requests have been enabled by a prepare I/O command, this decoded character presents an interruption request to the processor (a character-received request). At this time the controlling program can interpret the circumstances leading up to the request and, if desired, issue a reset ACCA command to reject the IPL. If the ACCA is not reset at this time, and the ACCA receives the DS character, a system reset occurs in the System/7, the processor is put into the load state, and the ACCA enters IPL mode. IPL data that is received by the ACCA is translated the same as if the ACCA were operating in binary data mode, described previously.

Loading into System/7 storage begins at location 0000 and continues sequentially until terminated by the host. The System/7 load light is turned on during this time. If no errors are detected by the host, it terminates the IPL operation with the characters EOB and LRC to indicate a normal ending. If no errors are detected by the ACCA during the message, the ACCA transmits a positive response to the host and exits from IPL mode. At this time, the System/7 processor branches to location 0000 to begin program execution on priority level 3, and the ACCA remains in receive mode. If interruption requests are enabled by a prepare I/O command, receipt of an EOT character from the host presents an interruption request to the processor (ISW bit 13 and the summary status indicator are set). When the EOT character is decoded, the ACCA enters standby mode of operation.



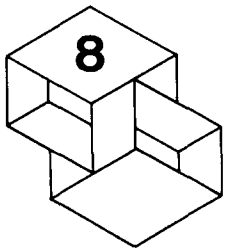
Notes:

Y Can be N if an error occurs

[1] ACCA must be in standby mode
[2] Interruption if prepared
[3] System reset if host attach switch is in enable and IPL position (ACCA in IPL mode)
[4] System reset and start program
[5] ACCA in standby mode
[6] Address

Figure 7-6. Initial program load operation

CHAPTER CONTENTS



FUNCTIONAL DESCRIPTION

The Binary Synchronous Communications Adapter (BSCA) is an optional feature that may reside in the 5010 Processor Module Models A or E.

The BSCA controls the transfer of serial data to and from another system at a remote location via a data set and communications line facility. The BSCA enables the System/7 to communicate in a synchronous communications mode with any of the following configurations, each having its own binary synchronous communications facility.

- System/7 to System/7
  (via point-to-point switched or leased lines only)
- System/7 to System/3
- System/7 to System/370 models 115 and 125
  (via System/370 integrated communications adapter)
- System/7 to System/370 Model 135
  (via System/370 Model 135 integrated communications adapter, IBM 2701, 2703, 3704, or 3705)
- System/7 to System/370 Model 145 or larger
  (via IBM 2701, 2703, 3704, or 3705)

Data to be transmitted or received by the System/7 is presented to the BSCA in EBCDIC or ASCII. EBCDIC data received by the System/7 is sent into storage without any code translation. For ASCII characters, the BSCA adjusts the ninth (parity) bit to conform to System/7 parity requirements.

The BSCA operates in half-duplex mode; that is, the System/7 or the host system can either transmit or receive data, but cannot transmit and receive simultaneously. Data rates are from 600 bits per second to 50,000 bits per second depending on the modem selected.

The following are the pluggable options for the BSCA feature.

- Modem has a ring indicator or no ring indicator
- a 2 or 4 wire line
- Program or nonprogram control of data terminal ready
- BSCA is a tributary on a multipoint network
- Point-to-point network
- System/7 provides the answer tone
- Multipoint hexadecimal address
- Internal clock speeds–600, 1200, 2000, or 2400 bits per second (BPS)

IBM MODEMS

The BSCA can communicate with another system over private lines, leased common-carrier facilities, or switched voice-grade common-carrier lines.

The following modems can be used with the BSCA.

IBM United States Modems

Modem	*Rate (bits/sec)*	*Line type*
3872	2400/1200	2 or 4 wire
3875	7200/3600	2 or 4 wire
Integrated	1200	2 or 4 wire

The pluggable options for the IBM 1200 BPS intergrated modem are as follows:

- Switch or leased line network
- Clear-to-send delay–25, 75, or 200 millisecond delay
- a 2 or 4 wire line
- Auto answer (optional feature for BSCA)

IBM World Trade Modems

3872
3875
3976 Model 3
3977 Model 2
3878 Models 12 and 14
5979 Models L11 and L12
1200 bps Integrated

In conformance with Electronic Industries Association (EIA) Standard RS232C within the United States, and Consultive Committee on International Telephone and Telegraph (CCITT) V24 Standards outside the United States, an attachment option provides the following interface signals:

1. Transmitted data
2. Received data
3. Request to send
4. Clear to send
5. Data terminal ready
6. Data set ready
7. Ring indicator
8. Transmit signal element timing
9. Receive signal element timing
10. Signal ground
11. Data signalling rate selector (wired for prime speed only)

Programming note: For more detailed information on binary synchronous communications conventions, refer to *General Information Binary Synchronous Communications*, Order No. GA27-3004.

The following table shows the control characters and sequences for BSCA:

Name	*Mnemonic*	*EBCDIC*	*ASCII*
Start of heading	SOH	SOH	SOH
Start of text	STX	STX	STX
End of transmission block[1]	ETB	ETB	ETB
End of text[1]	ETX	ETX	ETX
End of transmission[1]	EOT	EOT	EOT
Enquiry[1]	ENQ	ENQ	ENQ
Negative acknowledge[1]	NAK	NAK	NAK
Synchronous idle	SYN	SYN	SYN
Data link escape	DLE	DLE	DLE
Intermediate block character	ITB	IUS	US
Initial program load[2]	IPL	DC1 DC1 ENQ.	
Even acknowledge[1]	ACK 0	DLE (70)	DLE 0
Odd acknowledge[1]	ACK 1	DLE/	DLE 1
Wait before transmit-pos. ack[1]	WACK	DLE,	DLE;
Mandatory disconnect[1]	DISC	DLE EOT	DLE EOT
Reverse interrupt[1]	RVI	DLE@	DLE ⟨
Temporary text delay	TTD	STX ENQ	STX ENQ
Transparent start of text[3]	XSTX	DLE STX	
Transparent intermediate block[3]	XITB	DLE IUS	
Transparent end of text[3]	XETX	DLE ETX	
Transparent end of transmission block[3]	XETB	DLE ETB	
Transparent synchronous idle[3]	XSYN	DLE SYN	
Transparent block cancel[3]	XENQ	DLE ENQ	
Transparent TTD[3]	XTTD	DLE STX	
Data DLE in transparent mode[3]	XDLE	DLE DLE	

Notes:

1. These control characters and sequences cause a COD (change of direction) interruption request to the System/7 after the required action has been completed, if the chaining flag is off.
2. IPL sequence is [SEL] DC1 DC1 ENQ. SEL (address) required on multipoint network only. Not applicable in ASCII format.
3. Transparent mode is not available in ASCII.

The function of each control character follows.

Mnemonic	*Function*
SOH or STX	Reset control mode and set the adapter to text mode.
ETB or ETX	Reset text mode with block check character (BCC) comparison.
EOT	End of transmission.
ENQ	Reset text mode without BCC transmission and comparison.
NAK	Negative response to a request for a reply, or to a block of heading, or a block of text in error.
SYN	Transmitted automatically by the adapter to establish and maintain synchronization.
DLE	Alert the adapter to test the next character for a defined control sequence. In nontransparent text mode, DLE is treated as data. In transparent text mode, the transmitter adds a second DLE after each data DLE. At the receiver, the first DLE is stripped off and does not enter storage or the BCC.
ITB	Included in the BCC; it causes the BCC to be sent or received.
IPL	Control characters to decode an IPL sequence.
ACK 0	Indicate affirmative acknowledgement to even blocks.
ACK 1	Indicate affirmative acknowledgement to odd blocks.
WACK	Indicate a temporary not ready to continue/receive condition.
DISC	Used on switched communication facilities only, to initiate a disconnect.
RVI	Reverse direction of data transfer.
TTD	Alert the receiving station to a temporary time delay.
XSTX	Turn off control mode and set the adapter to transparent text mode.
XITB	Same as ITB, but also turn off transparent text mode.
XETX or XETB	Same as ETB or ETX but also turn off transparent mode.
XSYN	Transmitted automatically by the adapter to establish and maintain synchronization in transparent text mode.
XENQ	Turn off transparent text mode.
XTTD	Alter the receiving station to a temporary time delay in transparent text mode.
XDLE	Alert the adapter to test the next character for a defined control sequence in transparent text mode. The transmitter adds a second DLE after each data DLE. At the receiver, the first DLE is stripped off and does not enter storage or the BCC.

Synchronization and Timing Information

The BSCA receives strobing pulses from the modem which establish and maintain bit synchronization. If the modem does not supply a strobe, the Internal Clock Feature must be installed to supply synchronization. Whichever form of bit synchronization is used, a specific series of characters precedes each transmission in order to establish character synchronization.

Leading Pad Characters

If the Internal Clock feature is being used, BSCA automatically begins transmission with two leading pad characters (Hex 55) followed by the initial synchronizing pattern of two SYN characters.

Character Synchronization

Character synchronization is established by transmitting SYN SYN at the start of each transmission.

When the BSCA is transmitting in text mode, a sync pattern of SYN SYN is inserted at every transmit timeout. This action is used to maintain character synchronization.

Trailing Pad Characters

A trailing pad character (Hex FF) automatically follows every COD character (or after the BCC if the change of direction calls for BCC). This is done to assure that the last character sent (COD or BCC) goes on line in its entirety. A pad of FF also provides the second character of the NAK and EOT control character sequences. The BSCA does not begin an interrupt or chaining operation until the entire pad character is transmitted or received.

SYN and pad characters (leading and trailing) are handled by the attachment and are not stored in main storage.

BSCA OPERATING MODES

The BSCA operating modes, selected by control characters, are as follows.

Text Mode

This mode is selected when the first SOH or STX control character is decoded during a transmit or receive operation. In the text mode, the BSCA recognizes control characters and sequences separately from the data (text).

Transparent Text Mode

This mode is selected when a DLE STX control character is decoded during a transmit or receive operation. In this mode, the BSCA recognizes individual control characters and sequences only as data, with no other function. Any kind of binary data can be transmitted or received in this mode.

Monitor Mode

In this mode, the BSCA monitors all activity on the line such as entering text mode or transparent text mode. No data is transferred into storage and no data is transmitted. Monitor mode is entered after any of the following:

1. The host attach switch on the System/7 console is in the enable and IPL position and the BSCA is not in transmit mode. This allows monitoring for an IPL sequence.
2. A receive operation is accepted with the host attach switch in either the enable, or enable and IPL position.
3. A reset with the host attach switch in the enable and IPL position.

Monitor mode continues during control mode but ends when selected mode is entered.

Control Mode

This mode is entered when a valid EOT sequence is received. In control mode the BSCA monitors for its station address. A decoded SOH or STX control character discontinues control mode and returns the BSCA to monitor mode.

Selected Mode

This mode is entered when the BSCA has decoded its multi-point station address twice contiguously after establishing byte synchronization.

I/O COMMANDS

The BSCA is programmed using the prepare I/O command (as described in Chapter 4 under "Input/Output Instruction") and various immediate read and immediate write commands. The setting of the modifier field bits in the immediate commands further define the operations to be performed.

All I/O commands to the BSCA must have a device address of 001100 and a module address of 000000.

BSCA has its' own cycle steal function which operates independently of the optional cycle steal feature. Messages are transferred to and from BSCA by cycle steal cycles using parameters established by the BSCA device control block.

Device Control Block

The device control block (DCB) is an eight word area in storage set up by the user's program and contains the information needed to perform a BSCA cycle steal operation.

Each receive or transmit operation starts with the cycle steal fetch of the DCB. The address of the DCB is placed in the BSCA SAR by two PIO immediate write commands. The first command, write address, places the low order byte of the DCB address in positions 8 through 15 of the BSCA SAR. An initiate I/O command then places the high order byte of the address into the BSCA SAR to complete the address. At the completion of the initiate I/O command, the BSCA becomes busy and begins to cycle steal the DCB from main storage into various BSCA registers. Following the last DCB cycle, cycle steal operations continue depending on the active control word bits.

	0 15	
Word 0	Reserved	A
Word 1	Reserved	A + 1
Word 2	Reserved	A + 2
Word 3	Control word	A + 3
Word 4	DCB chain address	A + 4
Word 5	X DCB chaining flag (bit 1)	A + 5
Word 6	Word count	A + 6
Word 7	Starting data address	A + 7

A—the starting address in storage of the DCB. This address is established through the execution of the write address and the initiate I/O commands.

Word count and starting data address are not changed by the input/output operation.

DCB Word 3 (Control Word)

Bit 0-Receive Operation. When byte synchronization is established the BSCA begins transferring data into System/7 storage. A normal end interruption request to the System/7 processor is generated:

- When change of direction (COD) character is received and the chaining flag (described later in this section) is off.
- When the word count becomes 0 and the chaining flag is off.

If the data set ready line from the modem is off, or if transmit mode is on when this operation begins, an error interruption occurs immediately, setting on bit 1 (command reject) in the interrupt status word (ISW). Transmit mode will be on if the previous operation was a transmit operation and a COD character was not sent.

Bit 11 (start 3-second timer) may be used with this bit to limit the time allowed by the BSCA to establish character synchronization. Failure to establish synchronization within this time causes an interruption, setting on bit 3 (timeout) in the ISW.

Bit 1-Reserved.

Bit 2-Transmit Operation. A 3-second timeout starts, and request to send to the modem is turned on. When clear to send from the modem comes on, the timeout is turned off and the BSCA begins transferring data from System/7 storage. A normal channel-end/device-end interruption request to the System/7 processor is generated when:

- A COD character is transmitted and the chaining flag (described later in this section) is off.
- When the word count becomes 0 and the chaining flag is off.

If the data set ready line from the modem is off, or if receive mode is on while attempting to transmit, an error interruption occurs immediately, setting on bit 1 (command reject) in the ISW. Receive mode will be on if the previous operation was a receive operation and a COD character was not received.

Failure to receive clear to send from the modem within the 3-second timeout period causes an interruption, setting on bit 3 (timeout) in the ISW. Transmit mode is turned off and request to send to the modem is discontinued after the PAD character (eight 1-bits) is sent following a COD character or block check character (BCC) if required.

Bit 3-Allow Transparent Control. The ability to transmit control sequences while in transparent text mode is provided by setting on this bit. It should be set on in the final DCB used for a transparent text operation. It can be used only in a DCB following a block of transparent text.

If the BSCA is not in transparent text mode when this bit is presented, an error interruption occurs setting on bit 1 (command reject) in the ISW.

Programming note. It is suggested that bit 3 be used only with a word count of 1 (transparent end of text-DLE ETX; transparent end of transmission block-DLE ETB, etc.) to avoid unpredictable results should a data character and a DLE character have the same bit configuration.

Bit 4-Enable Terminal. If the host attach switch on the System/7 console is in the enable position, setting on this bit sets the data terminal ready and enables detection of data set ready. In a switched network, if the host attach switch is in the enable and IPL position, a ring indication also sets the data terminal ready. In a non-switched network, the data terminal is ready at all times.

Bit 11 (start 3-second timer) may be used with bit 4 to limit the time allowed by the BSCA for the data set to get ready. If the data set is not ready within this time, an interruption occurs, setting on bit 3 (timeout) in the ISW.

Bit 5-Disable Terminal. Setting on this bit discontinues the data terminal ready line to the data set, to allow disconnecting from a switched network. During this period (3 seconds) the BSCA is busy. At the completion of the disconnect, the BSCA causes an interruption if the chaining flag in word 5 of the DCB is off. Data terminal ready cannot be discontinued on a leased line.

Note: If the System/7 is dialed up erroneously, or if a switched line connection is interrupted during BSCA communications, then the data terminal ready line to the data set must be discontinued before line connection can be re-established. There are 3 ways of doing this:

1. Set on bit 5 to disable the terminal.
2. Put the host attach switch on the operator console in the disable position momentarily.
3. An automatic internal timer disables the terminal after 2.5 minutes.

Bit 6-Reserved.

Bit 7-Parity 1. This bit is used to maintain even parity in the high order byte of this control word.

Bit 8-Set ASCII Mode. Setting this bit on allows communication using ASCII code. When off, the BSCA recognizes EBCDIC code.

Bit 9-Reserved.

Bit 10-Start 2-second Timer. This bit is used to start a 2-second interval after which the BSCA interrupts. Timeout (bit 3) in the ISW is not set on.

Bit 11-Start 3-second Timer. This bit is used to start a 3-second interval after which the BSCA interrupts. Timeout (bit 3) in the ISW is not set on. This bit may also be used with bit 0 and/or with bit 4 to limit time.

Bits 12, 13, and 14-Reserved.

Bit 15-Parity 2. This bit is used to maintain even parity in the low order byte of this control word.

Following are the valid bit combinations for use in word 3. A parity error detected in word 3 causes an interruption, setting on bit 1 (command reject) in the ISW. Correct (even) parity in word 3 with an invalid bit combination causes unpredictable results.

General Control Functions

Hex	*Function*
0900	Enable
0911	Enable with 3-second timeout
0500	Disable
0021	2-second timeout
0011	3-second timeout

EBCDIC Control Functions

Hex	*Function*
8100	Receive
8111	Receive with 3-second timeout
2100	Transmit
1100	Transmit in transparent text mode

ASCII Control Functions

Hex	*Function*
8181	Receive
8190	Receive with 3-second timeout
2181	Transmit

DCB Word 4 (Chain address)

This word contains the address of the DCB to be used (chained to) when the current byte count is decremented to 0 or a COD character is recognized, and the chaining flag (bit 1) is on in word 5.

DCB Word 5 (Chaining flag)

Bit 0-Reserved. This bit is not used by the BSCA but must be set to 0.

Bit 1-Chaining Flag. When on, this bit indicates that word 4 contains the address of the chained-to DCB.

Bits 2-15. Not used by the BSCA but must be 0.

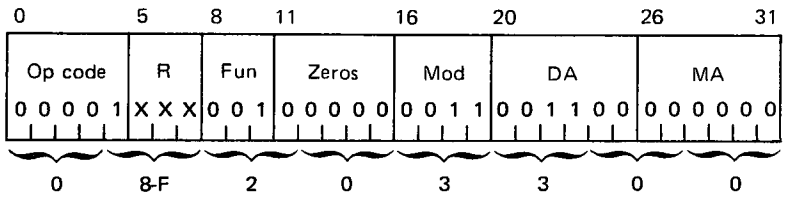
DCB Word 6 (Word count)

This word contains a count of the number of words to be transmitted or received.

DCB Word 7 (Starting data address)

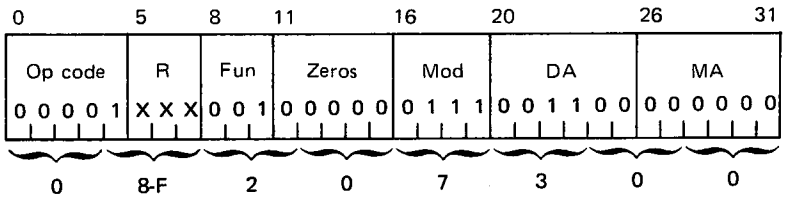
This word contains the starting address in System/7 storage to be used in fetching or storing data for a transmit or receive operation. After the DCB has been placed in the various BSCA registers, this address is placed in the BSCA SAR which determines the storage address of the data being handled by the BSCA. On transmit operations, one word is taken from storage by a cycle steal operation and is transmitted as two consecutive characters. On receive operations, two characters are received and stored in a buffer. Then a cycle steal operation is initiated to transfer them to storage as one word. If an odd number of characters is received, zeros are loaded into the low order byte.

Write Address



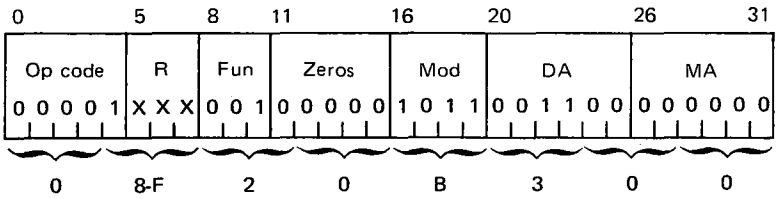
This command stores the contents of bits 8–15 from the register specified by the R field, or the accumulator if R=000, into the BSCA storage address register (SAR) bits 8–15, to form a partial DCB address. Condition code 2 is set if the BSCA is busy or an interruption is pending. This command does not cause an interruption.

Initiate I/O



This command stores the contents of bits 8 to 15 from the register specified by the R field, or the accumulator if R= 000, into the the BSCA storage address register (SAR), bits 0 to 7, to complete the DCB address. Condition code 2 is set if the BSCA is busy or an interruption is pending. If the BSCA has not been prepared, condition code 1 is returned and command reject (bit 1) is set in the direct control channel status word. If the BSCA has been prepared, the DCB is fetched from storage by the adapter. Subsequent action depends on the control bit(s) in word 3 of the DCB as described earlier in this chapter.

Write Diagnostic Control



This command uses bits 8 to 15 of the register specified by the R field or the accumulator if R=000, to perform various functions in diagnostic mode. It can be used to simulate transmit and receive conditions; to control the BSCA so that problems can be diagnosed.

Register Bits

8 Serial receive data. This bit is applied to the input of the shift register to simulate receive data.

9 Bit step. This bit steps a binary trigger that simulates the clocks received from a modem or internal clock card. Trigger output advances the bit ring which advances the shift register, BCC register, and other associated hardware. Two write diagnostic control commands (with this bit on) are needed to advance the bit ring by one.

10 Internal clock step. This bit simulates the 384 KHz oscillator when internal clock is installed. To advance the transmit/receive clock by one, the number of write diagnostic control commands (with this bit on) needed are:

1280 for 600 bits per second
640 for 1200 bits per second
384 for 2000 bits per second
320 for 2400 bits per second

11 Reset diagnostic operation. Diagnostic mode is reset when a diagnostic command is issued with this bit on.

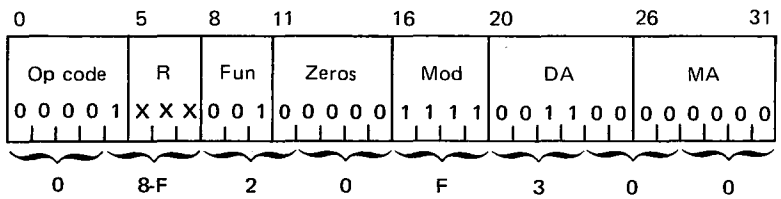
12 Ring indicator. This bit is used to simulate a "ring" from a switched line.

13 Set interrupt request. This bit set on initiates a BSCA interruption in order to isolate the interruption mechanism from normal BSCA controls.

14 Inhibit cycle steal clock. With this bit on, the program can issue write address and initiate I/O commands without the DCB being fetched. This permits the program to read the residual address from the BSCA storage address register and verify that the contents are as issued by the write address and initiate I/O commands.

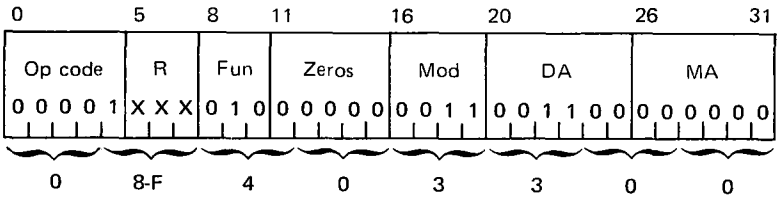
15 Set multipoint selected latch. This bit is used to simulate a selection sequence for testing the multipoint feature.

Reset BSCA



This command resets the BSCA except for the prepare information and the SAR. If the host attach switch on the System/7 console is in the enable and IPL position, monitor mode is entered. This command does not cause an interruption. The R field in this command is ignored.

Read ISW



This command reads the ISW into the register specified by the R field, or the accumulator if R= 000. It does not cause an interruption.

Interrupt Status Word (ISW)

Bits in the ISW are set on to indicate the operating status of the BSCA and to notify the system of errors detected. Execution of the read ISW command, after the BSCA recognizes that its request has been accepted by the processor, resets the ISW. The ISW is also reset by system reset, or the first new selection of the BSCA (not read ISW) after its interruption request has been accepted by the processor.

A description of each bit in the BSCA ISW follows.

Bits *Meaning*

0* Attention. This bit is set when the ring indicator line from the modem is active. It is not an error condition.

1* Command reject. This bit is set on by any of the following conditions:

1. The data set ready line is not active when either a transmit or receive operation is initiated.
2. A transmit operation is attempted while in receive mode or a receive operation is attempted while in transmit mode.
3. An allow transparent control operation is attempted while not in transparent text mode.
4. An odd number of bits (parity error) is word 3 of the DCB.
5. No function is requested in word 3 of the DCB.
6. On a multipoint network, a transmit operation is attempted before the status is selected.

2* Overrun. During a transmit operation, overrun occurs if no data is presented for transmission and synchronous idle control characters have been inserted for 3 seconds. During a receive operation, overrun occurs if the word count has been decremented to zero, no COD character was received, and another receive operation was not initiated by the time the receive buffer was filled.

In either case, the BSCA interrupts with this bit set on the next time it becomes busy.

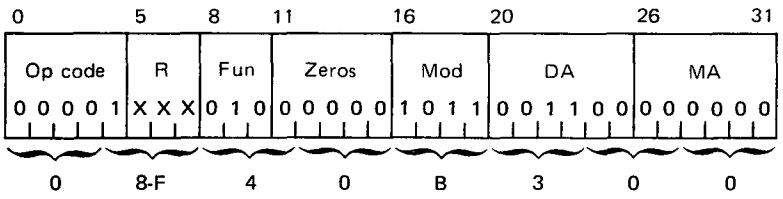
3* Timeout. Occurs under any of the following conditions:

1. Character phase (SYN SYN) is not established within 3 seconds after acceptance of a receive data operation. This is under program control and is effective only if bit 11 is set on in word 3 of the current DCB.
2. Continuous synchronization pattern, or transparent synchronization idle in transparent mode, is received for 3 seconds.
3. While receiving data, no synchronization pattern or no transparent synchronization idle is received for 3 seconds.
4. Clear to send is not returned from the modem within 3 seconds after request to send is presented.
5. Data set ready is not returned from the modem within 3 seconds after an enable with timer operation is begun.

Bits	*Meaning*
4*	No COD character. No turnaround character or sequence has been decoded by the BSCA, the word count has been decremented to zero, and the chaining flag is off. An EOT or NAK must be received with at least 4 contiguous one-bits following it to be valid.
5	Diagnostic mode. This bit is set on when the BSCA is in diagnostic mode due to a write diagnostic control command.
6	Data set ready. The modem has returned data set ready in response to data terminal ready to indicate the "off hook" condition during an auto answer operation. This bit is not reset by a read ISW command. Data set ready causes an interrupt only when the data set becomes ready in response to an enable operation and the chaining flag is off.
7*	Data set error. While in a transmit or receive mode the data set ready line from the modem has dropped with data terminal ready on, or the clear to send line from the modem has dropped with request to send on.
8*	Storage data check. The storage location accessed by the BSCA during the current instruction contains a parity error. The oepration is terminated and the data is not transmitted over the data link. The storage address register is not incremented.
9*	Program check. The main storage address presented by the BSCA during the current input/output operation for data, or DCB address, exceeds the storage size installed in the system. The operation is terminated. (An address of FFFD, FFFE, or FFFF presented as a DCB address does not cause a program check when the DCB is fetched.)
10	Reserved.
11	Channel end. This bit is set on at the termination of a subchannel busy condition. The busy condition exists from the initialization of a successful initiate I/O command to the point where the count goes to zero and the DCB chaining flag is off; a COD is detected with the chaining flag off; or an error or exception condition occurs. Channel end is not itself an error indicator; however, it is set on if any error terminates any subchannel operation. The BSCA does not separate channel-end interruptions from device-end interruptions.
12	Busy, This bit is set on when the BSCA accepts a data transfer operation and is ready to fetch the DCB, or the BSCA is given a disable terminal operation. It is not an error condition; it indicates that the BSCA is in the working state. Busy is reset when device end is set. Busy does not cause an interruption request.
13	Device end. This bit is set on whenever any operation is terminated, for whatever reason. It is not itself an error condition.
14*	Interface data check (BCC Error). The BCC received over the data link does not compare with the BCC accumulated. Also, LRC or VRC error in ASCII mode.
15	Multipoint. The BSCA is being used as a tributary in a multipoint configuration.

*The summary status bit is also set along with this bit.

Read ISW Extension (for diagnostic use only)

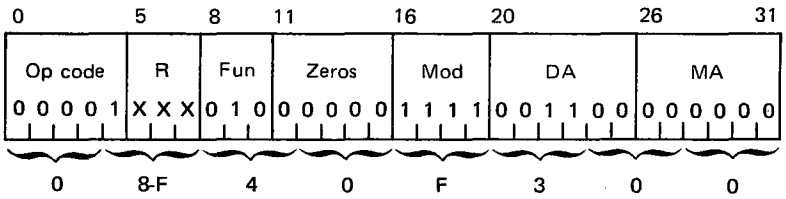


This command transfers 16 bits of BSCA status information for diagnostic use to the register specified by the R field, or the accumulator if R=000.

Register Bits (ON status)

0 *Bit advance* to the bit counter, clocks, shift register, etc. is active.
1 IPL mode. Set on by *xmit IPL* and stays on as long as the adapter is in IPL mode.
2 IPL start is sent to the CPU.
3 A SYN character has just been decoded.
4 An ACK0, WACK, or RVI has just been decoded.
5 A DLE has just been decoded.
6 Terminal address has been properly received (multipoint only).
7 Adapter in ASCII mode. If this bit is off, the adapter is in EBCDIC mode.
8 Status of the *serial receive data* line.
9 *Character phase* is active.
10 *Accum BCC* is active.
11 *Bit time 7* is active.
12 *Transparent mode* is active.
13 *Data transfer mode* is active.
14 *MST clear to send* is active.
15 Reserved.

Read Residual Address



This command stores the contents of the BSCA's storage address register (which is used during BSCA cycle steal operations and is updated by the BSCA) in the register specified by the R field, or the accumulator if R=000. Its purpose is to allow the program to determine the exact point during data transfer at which a subchannel error or exception condition occurred. This command does not cause an interruption. The storage address register in the BSCA is not reset by this command or by a reset I/O command; it is reset only by a power-on reset. Depending on the time the storage address register is read, it contains either the address of the next DCB to be used (in a chaining operation); the address of one of the words in a DCB; or the address used for data transfer.

INITIAL PROGRAM LOAD (IPL)

The host system determines when to initiate an IPL to the System/7 through the BSCA. The IPL is accepted only when the host attach switch on the System/7 console is in the ENABLE and IPL position. In a multipoint configuration, the BSCA maintains byte synchronization and searches for its assigned address character. When it is detected, followed by a second address character and the IPL sequence (DC1 DC1) ENQ, the adapter responds with an acknowledgement (ACK0) and enters the selected mode. In a point-to-point switched network IPL sequence, the address characters are not necessary. The host system then transmits DLE STX, to put the BSCA into transparent text mode, followed by the IPL program. The program is placed in System/7 storage starting at address 0000.

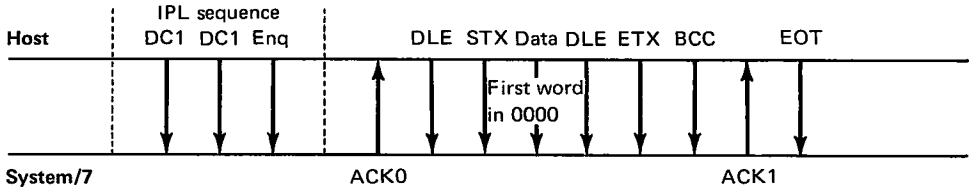
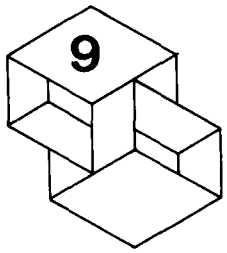


Figure 8-1. Initial program load operation

Upon receiving a COD control character followed by a valid BCC, the System/7 processor branches to location 0000 and begins executing the bootstrap program that processes the host IPL as a normal transparent message.

CHAPTER CONTENTS



The 5028 Operator Station (Figure 9-1) serves as the I/O device for operator communication with the system. In addition, it is used for initial program load (IPL) of the system from paper tape and for program preparation.

The operator station has a paper tape reader, a keyboard, and special control keys to enter data and control information. The operator station includes a paper tape punch and a printer to receive output from the system. Information is transferred from the system to the operator station at the rate of 10 characters per second. Information is transferred to the system from the paper tape reader at the rate of 10 characters per second. The rate at which information is transferred to the system from the keyboard varies according to the operator, the operating program, and the mechanical characteristics of the keyboard. It is, therefore, less than 10 characters per second.

A program-controlled bell in the operator station can be used to audibly alert the operator. The bell rings once each time a unique character code (hex 87) is transmitted to the operator station.

Graphics on the operator station keyboard and printer use the American Standard Code for Information Interchange (ASCII code) as listed in Appendix D. However, all 256 binary combinations can be punched into paper tape and read from it.

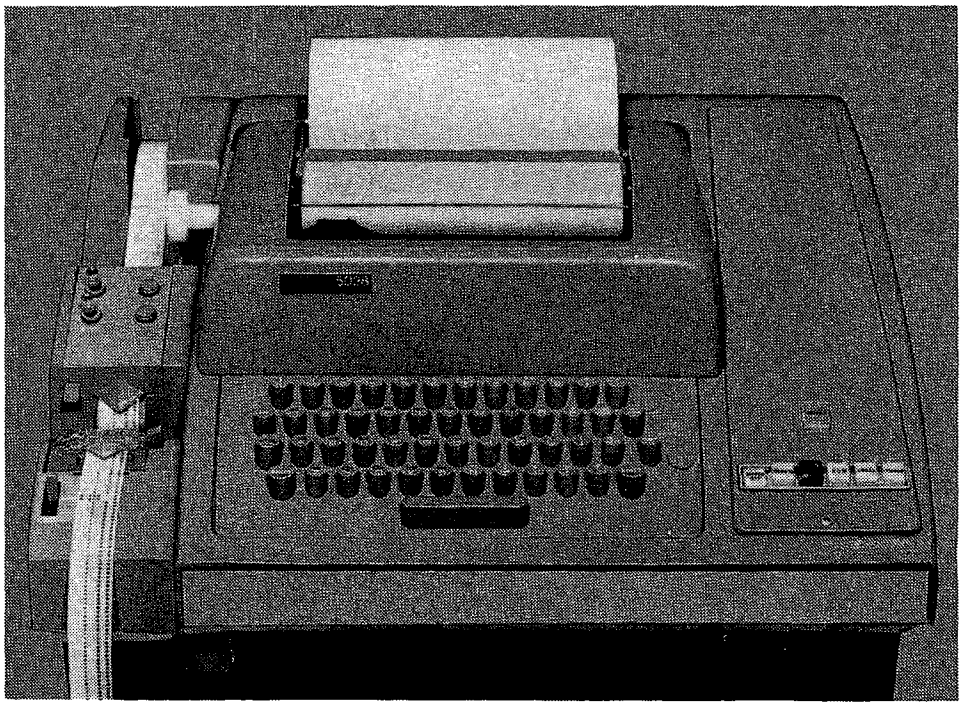
The printer uses 8-1/2-inch-wide (21,6 cm) roll paper. Characters are printed 10 per inch with a maximum of 72 characters per line horizontally, and 6 lines per inch vertically.

The printer is equipped with a sensing device to detect a low paper supply. When this occurs, following a print command, condition code 0 is returned and indicator bit 15 in the interrupt status word is set on.

The printing mechanism in the operator station automatically returns to the left margin after 72 characters have been printed. If a print command is directed to the operator station while the carriage is returning, the character may print in the middle of the page during carriage movement. To prevent this, the program should count the characters printed on any line and issue a carriage return command followed by a line feed command when the carriage approaches the end of the printing line. The carriage return and line feed commands should be in that order to give the carriage enough time to return to the left margin.

The tape punch uses one-inch-wide (2,54 cm) paper or Mylar* tape. Data is represented by eight character bits as shown in Figure 9-2. The tape punch is equipped with a sensing device to detect a low tape supply. When this occurs, following a punch command, condition code 1 is returned and the DCC status word bit 3 is set on.

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



BR1031A

Figure 9-1. IBM 5028 Operator Station

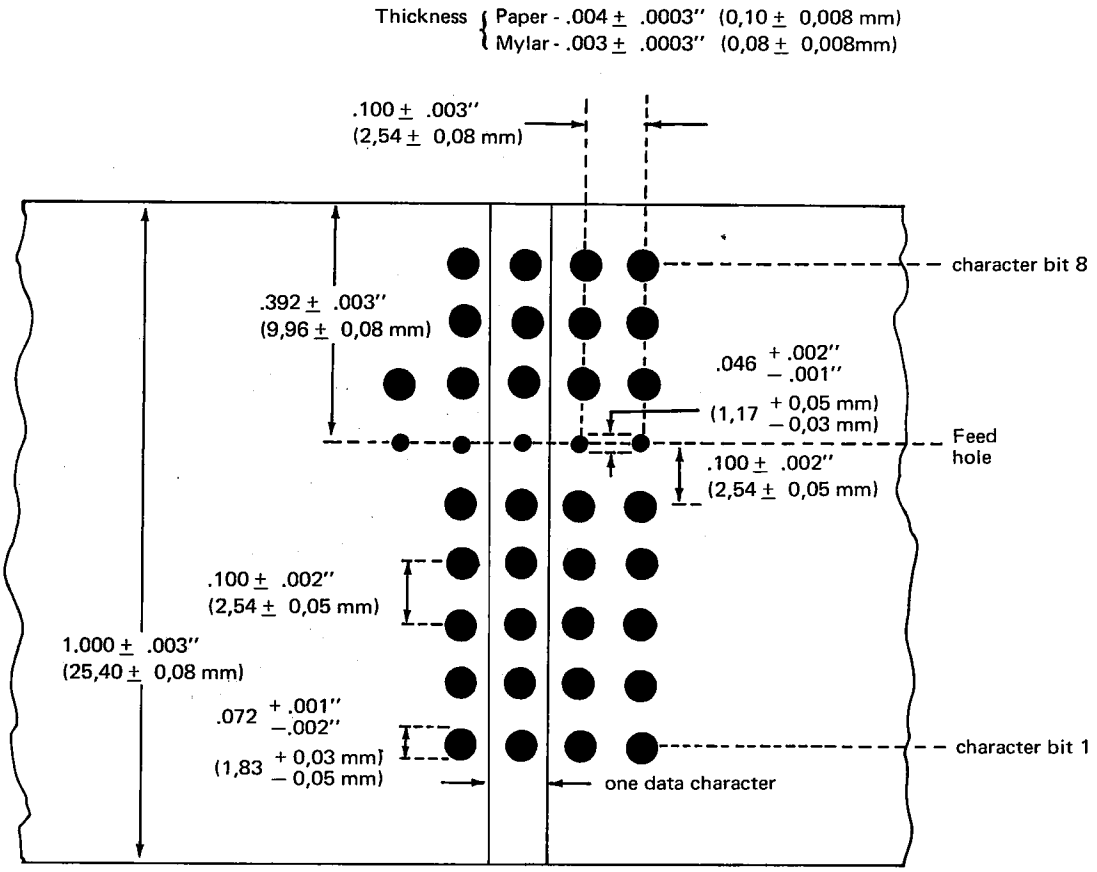


Figure 9-2. Punched tape data format

OPERATOR STATION FUNCTION AND COMMAND SWITCHES

Seven push-button switches are mounted on the operator station to the right of the keyboard. The functions of these switches are described in the following paragraphs.

Power

This switch controls power to the operator station. The switch is pressed once to turn on power and once to turn off power. The switch button lights when power to the operator station is turned on.

Motor Off

This switch turns off the motor in the operator station. Pressing the motor-off switch releases the remote or local switch if either was previously depressed.

Remote

This switch places the operator station in the remote mode of operation. When in remote mode, the operator station is controlled by the System/7.

Request

This switch serves a dual function in the operator station. When momentarily pressed once, the request switch generates an interruption request to the processor and sets indicator bit 0 in the operator station interrupt status word. A light inside the request switch turns on under program control when the operator station is in remote mode to indicate that the program requires information from the keyboard.

EOM (End of Message)

This switch, when momentarily pressed once, generates an interruption request to the processor and sets indicator bit 3 in the operator station interrupt status word. The keyboard must be enabled; that is, the request switch light is on.

Local

This switch places the operator station in the local mode of operation. When in local mode, the operator station is disconnected from the system and can be tested by the CE or used as a tape punch to prepare tapes for subsequent use.

The local switch also turns on the motor if it has been turned off by the motor-off switch.

Punch

This switch turns on or off the paper tape punch when the operator station is in local mode of operation. The switch is pressed once to turn on the punch, and once to turn off the punch. When the tape punch is turned on and the operator station is in local mode, any character read from the tape reader or entered from the keyboard is punched into paper tape.

INITIAL PROGRAM LOAD (IPL)

The two right-hand storage data/address switches must be set to a load unit address of hexadecimal 00 prior to performing an initial program load operation from the tape reader in the operator station.

When the program load key on the System/7 operator console is pressed, a system reset occurs and the operator station adapter turns on the motor, locks the keyboard, disables the printer, and feeds tape until a hole is detected (that is, a nonzero character is read).

The adapter then starts to transfer characters, two characters (bytes) at a time, in order to fill each 16-bit storage location word. Storing begins with location 0 and continues consecutively until location 127 is filled. (See Figure 9-3.)

The System/7 cannot be interrupted during this data transfer. Any error detected during this portion of the IPL procedure requires operator intervention. If an error occurs, the tape stops moving, but the load light on the system operator console remains on until reset by the reset key.

Following the transfer of 128 words of data, a branch is made automatically to location 0 and the system begins executing instructions on priority level 3. At this time, the tape stops moving, but the operator station motor remains on.

The IPL operation just described can also be started by the auto restart function. When the IPL operation is complete and program execution begins, bit 0 of the level 3 accumulator will be set off if the IPL was a result of the program load key. Bit 0 will be set on if the IPL was the result of an auto restart operation. (See "Auto Rstrt Switch" and "Program Load Key" in Chapter 5.)

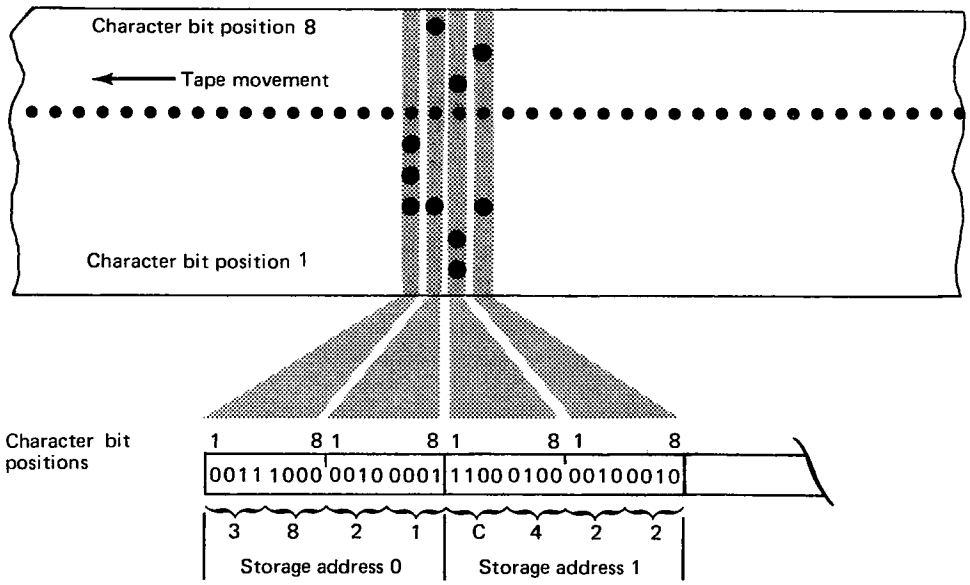


Figure 9-3. IPL from 5028 tape reader

I/O COMMANDS

The operator station is programmed using the prepare I/O command as described in Chapter 4 under "Input/Output Instruction." Data is transferred between the processor and the operator station adapter by immediate read and immediate write commands. The setting of the modifier field bits in an immediate command further defines the operation to be performed.

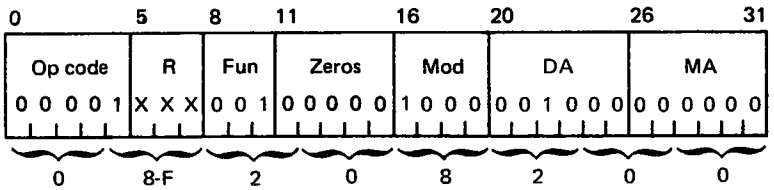
All I/O commands to the operator station must have a device address of 001000 and a module address of 000000.

Characters are stored into, or transmitted from, bits 8 to 15 of an index register or accumulator; bits 0 to 7 are not affected.

The four functions of keyboard entry, print, tape read, and tape punch are controlled by separate I/O commands.

Since the operator station is used for only relatively short periods of time, the motor in the operator station can be turned off and on by program control. The operator station command descriptions below specify if the motor must be started before issuing these commands. It is advisable to turn off the motor when the station will not be in use.

Turn On Motor

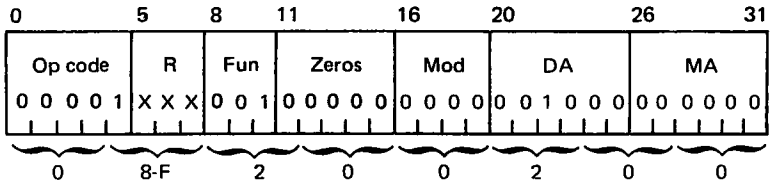


This command starts the operator station motor. An interruption request is presented to the system but the response time is the result of a complex relationship involving system programming, system hardware, and system operation. This interrelationship results in an interruption request that may occur from several microseconds to more than four seconds after the command has been issued. Therefore, the programmer is cautioned against using time-dependent code.

Note: This interruption request will occur even if the operator station motor is turned off at the operator station, or if the operator station power is turned off by the power switch.

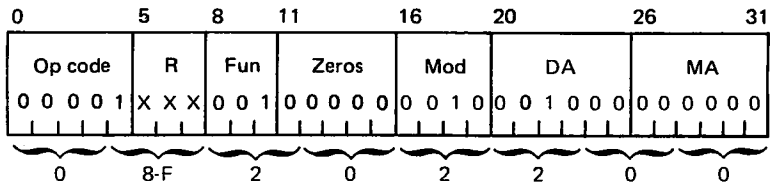
Condition code 2 is returned if the operator station adapter is busy or has an interruption pending. The contents of the R field are ignored.

Turn Off Motor and Lock Keyboard



This command stops the operator station motor and locks the keyboard. No interruptions occur as a result of this command. The contents of the R field are ignored.

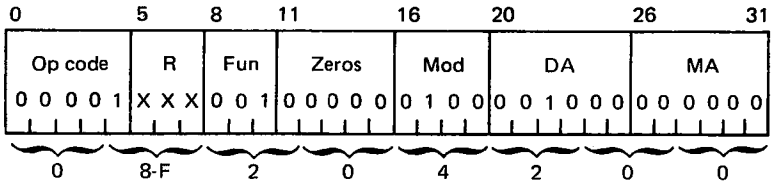
Print Only



This command sends one character to the operator station for printing. The single character is obtained from bits 8 to 15 of the index register specified by the R field, or from the accumulator if R=000. After the character is printed, the operator station adapter presents an interruption request.

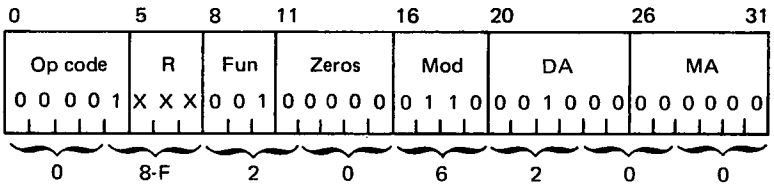
If the operator station motor is not running, condition code 1 is returned and bit 3 of the direct control channel status word is set on. Condition code 2 is returned if the operator station is busy or has an interruption pending.

Punch Only



This command sends one character to the operator station for punching on tape. The single character is obtained from bits 8 to 15 of the index register specified by the R field, or from the accumulator if R=000. Tape movement through the punch mechanism is automatic, stopping after each character is punched. The operator station adapter presents an interruption request after the character is punched.

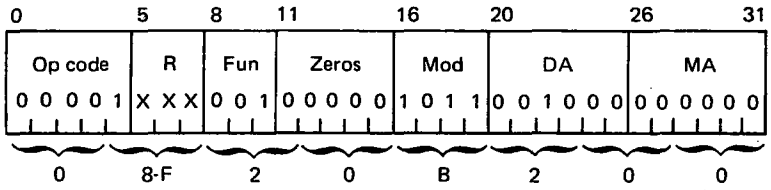
Print and Punch



This command sends one character to the operator station for both punching on tape and printing. The single character is obtained from bits 8 to 15 of the index register specified by the R field, or from the accumulator if R=000. Tape movement through the punch mechanism is automatic, stopping after each character is punched. The operator station adapter presents an interruption request after the character is punched and printed.

If the operator station motor is not running, condition code 1 is returned and bit 3 of the direct control channel status word is set on. Condition code 2 is returned if the operator station is busy or has an interruption pending.

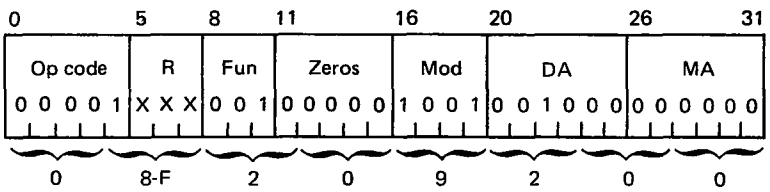
Keyboard Entry and Print



This command unlocks the keyboard, enables the printer, and turns on the operator station request switch light. When an operator enters a character on the keyboard, the character is printed and its character code is transmitted to the operator station adapter. The keyboard is then locked, the printer is disabled, the request switch light is turned off, and the operator station adapter presents an interruption request. After the interruption request is presented, a read command must be used to obtain the character from the adapter. Following this read command, another keyboard-entry-and-print command is normally issued to accept the next character from the keyboard. The contents of the R field are ignored in the keyboard-entry-and-print command.

If the operator station motor is not running, condition code 1 is returned and bit 3 of the direct control channel status word is set on. Condition code 2 is returned if the operator station is busy or has an interruption pending.

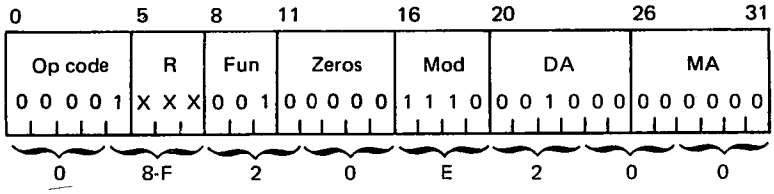
Keyboard Entry and No Print



This command performs the same function as the keyboard-entry-and-print command, except that the character is not printed. The command unlocks the keyboard and turns on the operator station request switch light. When the operator enters a character on the keyboard, the character code is transmitted to the operator station adapter. The keyboard is then locked, the request switch light is turned off, and the adapter presents an interruption request. After the interruption request is presented, a read command must be used to obtain the character from the adapter. Following this read command, another keyboard-entry-and-no-print command is normally issued to accept the next character from the keyboard. The contents of the R field are ignored in the keyboard-entry-and-no-print command.

If the operator station motor is not running, condition code 1 is returned and bit 3 of the direct control channel status word is set on. Condition code 2 is returned if the operator station is busy or has an interruption pending.

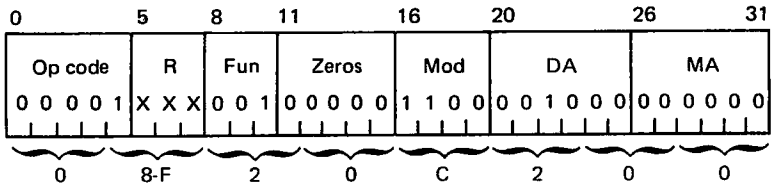
Feed Tape and Print



This command enables the printer and moves the tape to the next character. This character is printed, and causes an interruption request. The contents of the R field are ignored in the feed-tape-and-print command.

If the operator station motor is not running, condition code 1 is returned and bit 3 of the direct control channel status word is set on. Condition code 2 is returned if the operator station is busy or has an interruption pending.

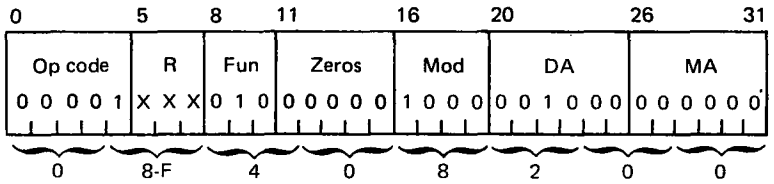
Feed Tape and No Print



This command transfers the character under the paper tape pin feed to the operator station adapter. The tape is then moved to the next character, and an interruption request is presented. After the interruption request is presented, a read command must be used to obtain the character from the adapter. The contents of the R field are ignored in the feed-tape-and-no-print command.

If the operator station motor is not running, condition code 1 is returned and bit 3 of the direct control channel status word is set on. Condition code 2 is returned if the operator station is busy or has an interruption pending. This command transfers the first character sensed from tape into the operator station adapter. Subsequent characters and tape feed cycles are obtained by the read-character-with-tape-feed command alone. The sequence is stopped by the read-character-without-tape-feed command.

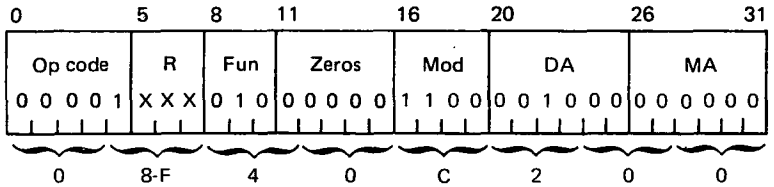
Read Character With Tape Feed



This command stores a single character from the operator station adapter into bits 8 to 15 of the index register specified by the R field, or the accumulator if R=000. The character under the paper tape pin feed is transferred to the operator station adapter. The tape is then moved to the next character, and an interruption request is presented. After the interruption request is presented, a read command must be used to obtain the character from the adapter.

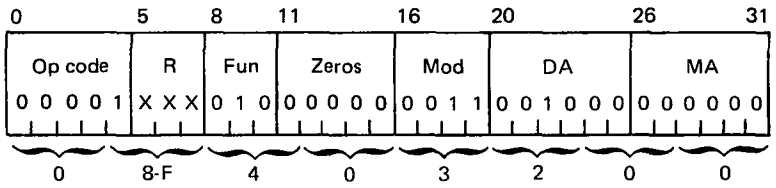
If the operator station motor is not running, condition code 1 is returned and bit 3 of the direct control channel status word is set on. Condition code 2 is returned if the operator station is busy or has an interruption pending.

Read Character Without Tape Feed



This command stores a single character from the operator station adapter into bits 8 to 15 of the index register specified by the R field, or the accumulator if R=000. No interruptions occur as a result of this command. Condition code 2 is returned if the operator station is busy or has an interruption pending.

Read ISW



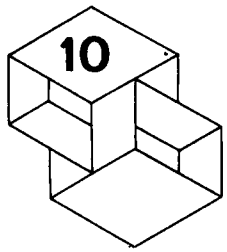
The read ISW command stores the operator station ISW into the index register specified by the R field, or the accumulator if R=000.

Interrupt Status Word (ISW): The 16 bits in the interrupt status word are set to indicate operating status of the operator station and to request service from the processor. Interruptions result from setting any ISW bit except the busy and printer paper low bits. If the ISW is successfully read, the ISW is reset if the interruption request is accepted by the system. The ISW is also reset by a system reset, or by the first new selection of the operator station by a command (not necessarily read ISW) after the interruption has been accepted by the system.

Significant bits in the ISW and their meanings are:

Significant Bits	*Meaning*
0	Attention. The operator station request key was pressed. If the operator station adapter is not busy and no interruption is pending, the adapter presents an interruption request. If the adapter is busy or has an interruption pending, the attention interruption bit will be presented in the ISW along with the device-end bit when the end interrupt occurs.
3	End of message. The EOM (end of message) key on the operator station was depressed. This status bit also sets ISW bit 13 and presents an interruption request to the processor.
12	Device busy. The operator station motor is reaching operating speed, or a command has been issued that will result in an interruption request, which has not yet occurred.
13	Device end. The operator station has performed the requested operation or requires service from the processor.
15	Printer paper low. Indicates that the paper supply in the 5028 is low. This bit is reset by adding more paper; it is not reset by a read ISW command.

CHAPTER CONTENTS



The 5014 Analog Input Module Model B1 converts analog signals into binary values of 14 bits plus sign. These analog signals can be within the range of ±10 mV full scale to ±5.12V full scale.

This analog input module can be located in any or all of the I/O module positions within an IBM 5026 Enclosure. (Location 00 in the 5026 is the processor module location.)

FUNCTIONAL DESCRIPTION

In addition to logic controls, the model B1 I/O module consists of an analog-to-digital converter (ADC), an amplifier, and a point-select relay multiplexer capable of a maximum scanning speed of 200 points per second. As many as 128 two-wire differential analog input points can be accommodated, in eight groups of 16 points each.

One amplifier is required to service all of the analog input points. If all analog inputs are at the ±5.12V level, the high-level amplifier is used. If a range of voltage levels must be accommodated, a multirange amplifier is used.

The multirange amplifier can be program controlled to one of seven different ranges, or to automatically select the appropriate amplifier gain. When the amplifier is programmed to automatically select the gain, analog signal resolution is reduced to 12 bits plus sign if the selected gain is returned (binary encoded in three bits) to the program with the data. The multirange amplifier allows conversion of signals in the ±10 mV, ±20 mV, ±40 mV, ±80 mV, ±160 mV, ±640 mV, and ±5.12V ranges. (See Appendix E for conversion of an analog voltage to its binary equivalent for the amplifier range selected.)

The analog-to-digital converter furnishes an overload indication in the data word when the signal level falls outside of a range that can be converted.

A temperature reference attachment is available for use in the cold junction compensation of thermocouple inputs.

The 5014 Model E2 provides the capacity to expand a Model B1 an additional 128 points, divided into eight 16-point groups. Two Model E2 modules may be attached to each Model B1. The first Model E2 module must be located directly under the Model B1. The second Model E2 module, if any, must be located directly under the first Model E2 module. Points in the Model E2 module are addressed through the Model B1 module.

I/O COMMANDS

The 5014 Analog Input Module Model B1 is programmed using the prepare I/O and halt I/O commands as described in Chapter 4 under "Input/Output Instruction." Immediate read and immediate write commands transfer data between the analog input module and the processor module. The setting of the modifier field bits in the command further defines the operation to be performed.

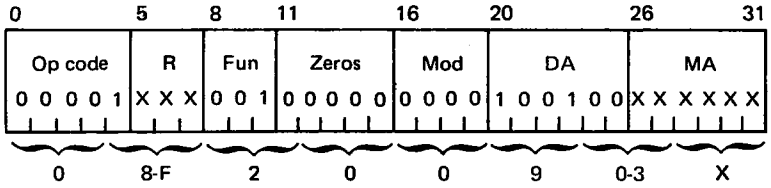
In all I/O commands addressed to the analog input module, the device address field is not used. However, for program compatibility with analog input control model B in a 5012 Multifunction Module, the device address field may contain 100100 for all commands except read ID and read DSW. The module address varies depending upon the physical location of the I/O module in the 5026 Enclosure.

Usually, a complete operation is performed within the time limit required for execution of the immediate commands. Exceptions to this rule are two convert commands that start the conversion cycle and then permit the program to resume normal operation while awaiting the interruption request presented by the module at the end of the conversion cycle.

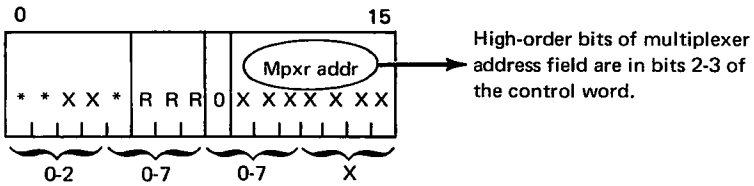
Executing either of the two convert commands sets on the ISW device-busy bit, which remains on during the conversion time. When the conversion is complete, an interruption request is presented to the processor, the device-busy bit is turned off, and the device-end bit is turned on. Condition code 2 is returned to all immediate commands issued (except read ISW or read DSW) while the module is busy or has a pending interruption.

The normal analog input programming routine is shown in Figure F-1 of "Appendix F."

Convert Analog Input



This command starts an analog-to-digital conversion cycle based upon control information contained in the index register (R), or the accumulator if R=000. The control information word has the format:



The range bits (RRR) in the control word specify the input voltage range and the amplifier gain desired, as shown in the following table.

RRR Bits	*Input Voltage Range*	*Amplifier Gain*
000	–	Automatic gain
001	±10 mV	512
010	±20 mV	256
011	±40 mV	128
100	±80 mV	64
101	±160 mV	32
110	±640 mV	8
111	±5.12V	1

When automatic gain is specified (RRR bits=000), the optimum gain (one of the seven amplifier gains) is selected automatically, based upon the input signal voltage. The selected gain can be returned to the program, along with the converted signal data, by issuing a read-ADC command. Since the indication of gain requires three bits, the input data itself is reduced to 12 bits plus sign.

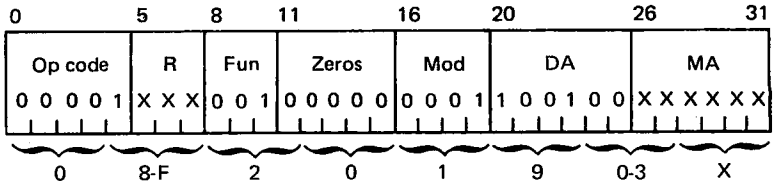
The range bits (RRR) have no significance if the high-level amplifier is installed.

This convert-analog-input command starts an analog-to-digital conversion cycle of the analog input point specified by the multiplexer address field in the control word. An interruption request occurs at the end of the conversion cycle. The results of the conversion are then obtained by issuing a read-ADC or read-ADC-extended-precision command.

With the read-ADC command, the data format is in short form (12 bits plus sign) if automatic gain was specified, or in standard form (14 bits plus sign) if a programmed *range* was specified. The read-ADC-extended-precision command always returns the data in standard form.

The multiplexer address field in the control word consists of bits 2-3 (the field high-order bits) and 9-15. The nine bits can be considered as a binary field to encode addresses 000 to 383. Bit 8 is reserved and set to zero. Bits 2-3 are necessary because an analog input configuration (models B1/E2/E2) can have up to 384 points.

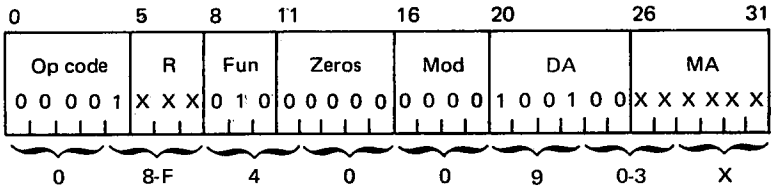
Convert Analog Input with External Synchronization



This command functions in the same manner as the convert-analog-input command except that the start of conversion depends upon the receipt of an external synchronization input pulse.

The user provides the external synchronization input pulse to start the analog-to-digital data conversion. (Refer to the *System/7 Installation Manual–Physical Planning,* Order No. GA34-0004, for more detailed data on the external synchronization input pulse.)

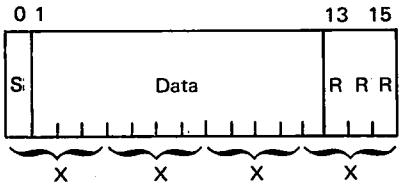
Read ADC



This command stores the binary output value from the ADC into the index register (R), or the accumulator if R=000.

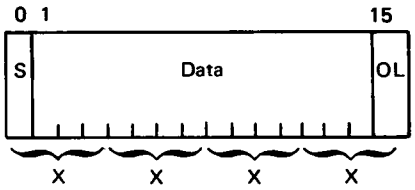
Data stored in the register can have either of two formats, depending upon whether the last convert command requested automatic gain or selected a particular amplifier gain.

If automatic gain was requested, the 16 bits of data have the format:



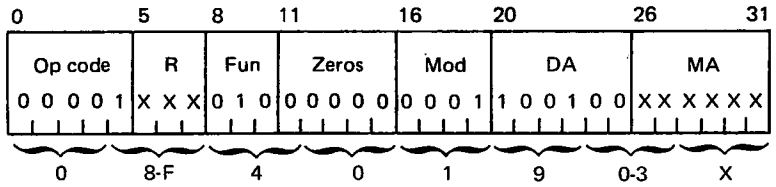
Twelve bits of binary data plus sign (S bit) are returned. The RRR field indicates which amplifier gain was automatically selected by the amplifier. (Refer to the table in the "Convert Analog Input" command description.) If RRR=000, an overload condition occurred with the automatic gain selection.

If a particular amplifier gain was selected by the last convert command, the 16 bits of data have the format:



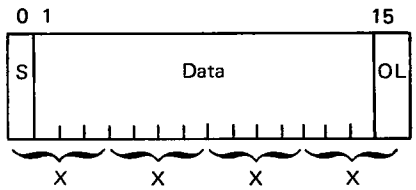
Fourteen bits of binary data plus sign (S bit) are returned. The single-bit OL field is set on if an overload condition occurred with the program-selected amplifier gain.

Read ADC Extended Precision



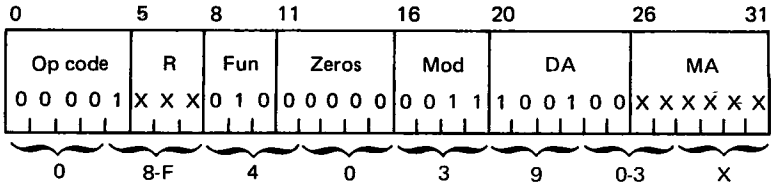
This command stores in standard form (14 bits plus sign) the binary output value from the ADC when automatic amplifier gain was selected by the last convert command.

The binary output value resulting from the last conversion cycle is stored in the index register (R), or the accumulator if R=000. The data transferred to the register has the format:



Fourteen bits of binary data plus sign (S bit) are returned. The single-bit OL field is set on if an overload condition occurred with the automatic amplifier gain selection. Note that the automatically selected gain is not returned with the data. This gain can be determined by issuing a read-ADC command before or after the read-ADC-extended-precision command.

Read ISW



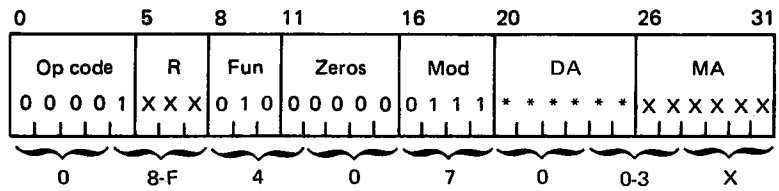
This command stores the 16-bit interrupt status word (ISW) associated with the analog input module into the index register (R), or the accumulator if R=000.

Interrupt Status Word (ISW): The 16 bits in the interrupt status word are set by conditions occurring after a convert-analog-input or a convert-analog-input-with-external-synchronization command is completed but while the device is still busy. Since these errors cannot be indicated to the program by the condition code, interruptions result from setting on any ISW bit except the busy bit. Setting on any one of bits 3 through 7 also sets on the summary status indicator when the resulting interruption request occurs. After the interruption is accepted, executing any command other than read DSW resets the ISW.

The significant bits in the ISW and their meanings are:

Significant Bits	*Meaning*
3	Analog-to-digital converter inoperative.
5	Multiple relays selected. This multiplexer failure is caused by more than one input being selected during the conversion cycle.
6	No relays selected. This multiplexer failure results from no input being selected during the conversion cycle.
7	Overload. The voltage read by the ADC was greater than the input voltage range selected by the amplifier.
12	Device busy. The analog input module is converting an input voltage.
13	Device end. The conversion operation requested is complete. If error conditions occur during the execution of I/O commands, this bit is set on along with other error identifying bits.
15	Invalid multiplexer address. An input point was addressed that was not installed.

Read DSW



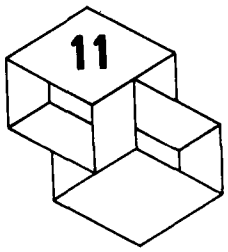
This command stores the 16-bit device status word (DSW) associated with the analog input module into the index register (R), or the accumulator if R=000. Execution of this command resets the bits in the DSW.

Device Status Word (DSW): The 16 bits in the device status word are set as a result of detected error conditions occurring during the execution of an immediate command to the I/O module. The operating program is notified of these error conditions by returning condition code 1 to the processor. If other commands are attempted before resetting the DSW bits, condition code 1 continues to be returned to the program.

The significant bits in the DSW and their meanings are:

Significant Bits	*Meaning*
1	Command reject. The analog input module cannot execute the command. (For example, an interruption-causing command issued to the device when it is disabled for interruptions.)
14	Data check. The analog input module has detected a parity error involving data. The operation that caused this condition usually can be retried successfully if the error condition is intermittent.

CHAPTER CONTENTS



The 5014 Analog Input Module Model C1 converts analog signals into binary values of 14 bits plus sign. These analog signals can be within the range of ±10 mV full scale to ±5.12V full scale.

This analog input module can be located in any or all of the I/O module positions within an IBM 5026 Enclosure. (Location 00 in the 5026 is the processor module location.)

FUNCTIONAL DESCRIPTION

In addition to logic controls, the model C1 I/O module consists of an analog-to-digital converter (ADC), an amplifier, and a differential solid-state multiplexer.

The solid-state multiplexer can have as many as 128 two-wire differential analog input points, in eight groups of 16 points each. With high-level inputs (±5.12V), a scanning speed of 20,000 points per second can be attained. A scanning speed of 14,000 points per second can be achieved with low-level inputs (10 to 640 mV). A scanning speed of 7,000 points per second can be obtained with the use of automatic gain selection. The actual scanning rate obtained, however, depends on the input voltage value, the physical location of the I/O module, and the programming techniques used. Conversion time required by the ADC varies from 25 to 46 microseconds, depending on the input voltage. Additional delays due to module location and programming must also be considered when computing the actual scanning rate.

One amplifier is required to service all of the analog input points. If all analog inputs are at the ±5.12V level, the high-level amplifier is used. If a range of voltage levels must be accommodated, a multirange amplifier is used.

The multirange amplifier can be program controlled to one of the seven different ranges, or to automatically select the appropriate amplifier gain. When the amplifier is programmed to automatically select the gain, analog signal resolution is reduced to 12 bits plus sign if the selected gain is returned (binary encoded in three bits) to the program with the data. The multirange amplifier allows conversion of signals in the ±10 mV, ±20 mV, ±40 mV, ±80 mV, ±160 mV, ±640 mV, and ±5.12V ranges. (See Appendix E for conversion of an analog voltage to its binary equivalent for the amplifier range selected.)

The analog-to-digital converter furnishes an overload indication in the data word when the signal level falls outside of a range that can be converted.

A temperature reference attachment is available for use in the cold junction compensation of thermocouple inputs.

I/O COMMANDS

The 5014 Analog Input Module Model C1 is programmed using the prepare I/O and halt I/O commands as described in Chapter 4 under "Input/Output Instruction." Immediate read and immediate write commands transfer data between the analog input module and the processor module. The setting of modifier field bits in the command further defines the operation to be performed.

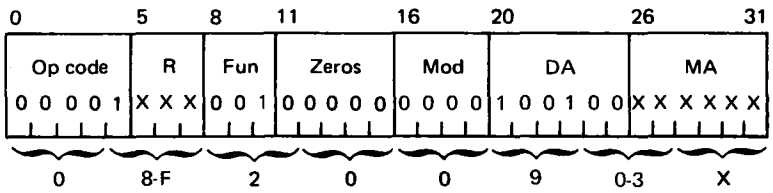
In all I/O commands addressed to the analog input module, the device address field is not used. However, for program compatibility with analog input control model C in a 5012 Multifunction Module, the device address field may contain 100100 for all commands except read ID and read DSW. The module address varies depending upon the physical location of the I/O module in the 5026 Enclosure.

Usually, a complete operation is performed within the time limit required for execution of the immediate commands. Exceptions to this rule are three convert commands that start the conversion cycle and then permit the program to resume normal operation while awaiting the interruption request presented by the module at the end of the conversion cycle.

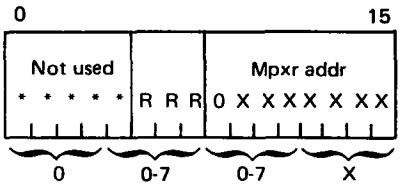
Executing any of the three commands sets on the ISW device-busy bit, which remains on during the conversion time. When the conversion is complete, an interruption request is presented to the processor, the device-busy bit is turned off and the device-end bit is turned on. Condition code 2 is returned to all immediate commands issued (except read ISW or read DSW) while the module is busy or has a pending interruption.

The normal analog input programming routine is shown in Figure F-1 of "Appendix F."

Convert Analog Input



This command starts an analog-to-digital conversion cycle based upon control information contained in the index register (R), or the accumulator if R=000. The control information word has the format:



The range bits (RRR) in the control word specify the input voltage range and the amplifier gain desired, as shown in the following table.

RRR Bits	*Input Voltage Range*	*Amplifier Gain*
000		Automatic gain
001	±10 mV	512
010	±20 mV	256
011	±40 mV	128
100	±80 mV	64
101	±160 mV	32
110	±640 mV	8
111	±5.12V	1

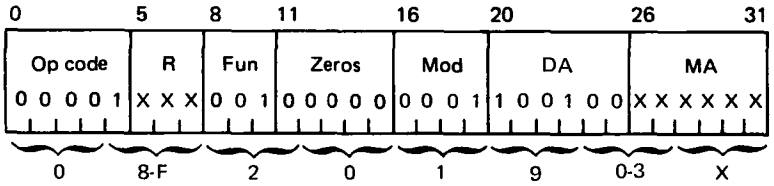
When automatic gain is specified (RRR bits=000), the optimum gain (one of the seven amplifier gains) is selected automatically, based upon the input signal voltage. The selected gain can be returned to the program, along with the converted signal data, by issuing a read-ADC command. Since the indication of gain requires three bits, the input data itself is reduced to 12 bits plus sign.

The range bits (RRR) have no significance if the high-level amplifier is installed.

This convert-analog-input command starts an analog-to-digital conversion cycle of the analog input point specified by the multiplexer address field in the control word. An interruption request occurs at the end of the conversion cycle. The results of the conversion are then obtained by issuing a read-ADC, a read-ADC-extended-precision, or a read-and-convert-ADC command.

With either the read-ADC command or the read-and-convert-ADC command, the data format is in short form (12 bits plus sign) if automatic gain was specified, or in standard form (14 bits plus sign) if a particular range was specified. The read-ADC-extended-precision command always returns the data in standard form.

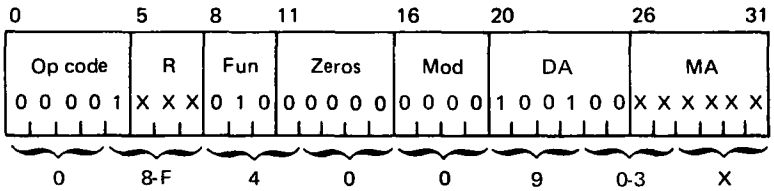
Convert Analog Input with External Synchronization



This command functions in the same manner as the convert-analog-input command except that the start of conversion depends upon the receipt of an external synchronization input pulse.

The user provides the external synchronization input pulse to start the analog-to-digital data conversion. (*System/7 Installation Manual–Physical Planning,* Order No. GA34-0004, contains more detailed data on the external synchronization input pulse.)

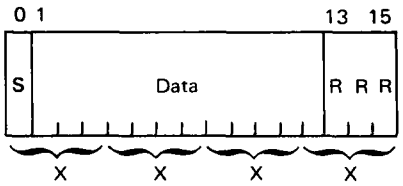
Read ADC



This command stores the binary output value from the ADC into the index register (R), or the accumulator if R=000.

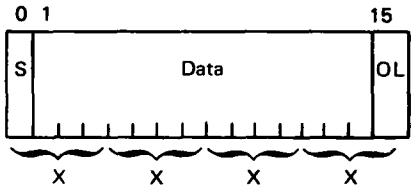
Data stored in the register can have either of two formats, depending upon whether the last convert command requested automatic gain or selected a particular amplifier gain.

If automatic gain was requested, the 16 bits of data have the format:



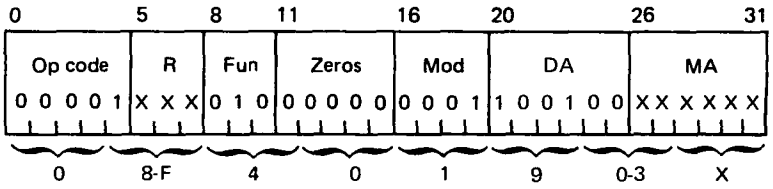
Twelve bits of binary data plus sign (S bit) are returned. The RRR field indicates which amplifier gain was automatically selected by the amplifier. (Refer to the table in the "Convert Analog Input" command description.) If RRR=000, an overload condition occurred with the automatic gain selection.

If a particular amplifier gain was selected by the last convert command, the 16 bits of data have the format:



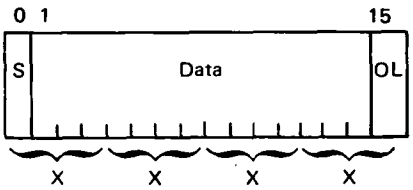
Fourteen bits of binary data plus sign (S bit) are returned. The single-bit OL field is set on if an overload condition occurred with the program-selected amplifier gain.

Read ADC Extended Precision



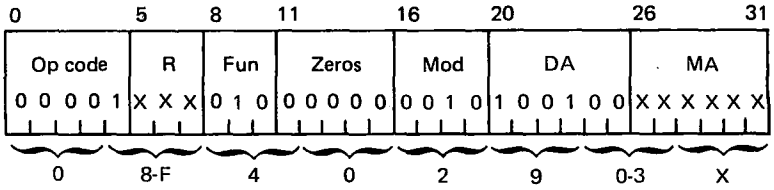
This command stores in standard form (14 bits plus sign) the binary output value from the ADC, when automatic amplifier gain was selected by the last convert command.

The binary output value resulting from the last conversion cycle is stored in the index register (R), or the accumulator if R=000. The data transferred to the register has the format:



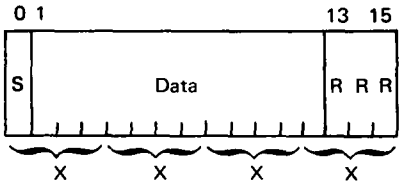
Fourteen bits of binary data plus sign (S bit) are returned. The single-bit OL field is set on if an overload condition occurred with the automatic amplifier gain selection. Note that the automatically selected gain is not returned with the data. This gain can be determined by issuing a read-ADC command before or after the read-ADC-extended-precision command.

Read and Convert ADC



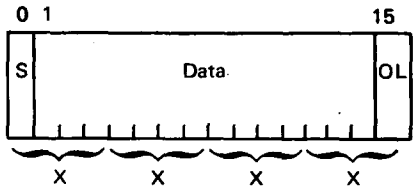
This command is used for repetitive operations on a single analog input point. The read-and-convert-ADC command stores the binary output value from the ADC into the index register (R), or the accumulator if R=000, and starts another conversion cycle.

Data transferred to the register can have either of two formats, depending upon whether the last convert command requested automatic gain or selected a particular amplifier gain. If automatic gain was selected, the 16 bits of data have the format:



Twelve bits of binary data plus sign (S bit) are returned. The RRR field indicates which amplifier gain was automatically selected by the amplifier. (Refer to the table in the "Convert Analog Input" command description.) If RRR=000, an overload condition occurred with the automatic gain selection.

If a particular amplifier gain was selected by the last convert command, the 16 bits of data have the format:



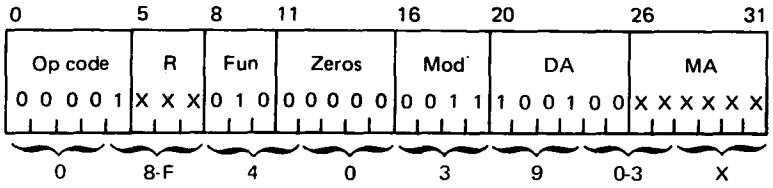
Fourteen bits of binary data plus sign (S bit) are returned. The single-bit OL field is set on if an overload condition occurred with the program selected amplifier gain.

The convert portion of the read-and-convert-ADC command converts the analog input point specified by the last convert command, utilizing the same amplifier gain. Thus, if automatic gain was specified by the last convert command, then automatic gain is also used by each subsequent execution of the read-and-convert-ADC command.

Since this command performs repetitive operations on a single analog input point, the solid-state multiplexer is monopolized during this time. To terminate this repetitive read sequence and free the multiplexer, a read-ADC command must be issued if the result of the last conversion cycle is needed by the program. However, any command other than the read-and-convert-ADC command may be issued to terminate the repetitive read sequence.

The conversion cycle started by a read-and-convert-ADC command does not depend on an external synchronization pulse, nor can it be started by an external synchronization pulse.

Read ISW



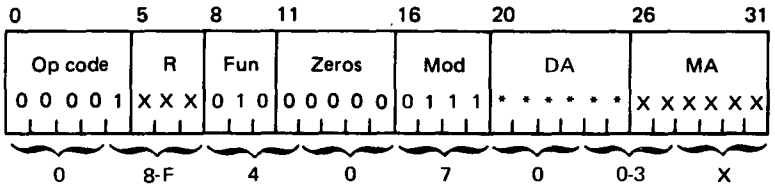
This command stores the 16-bit interrupt status word (ISW) associated with the analog input module into the index register (R), or the accumulator if R=000.

Interrupt Status Word (ISW): The 16 bits in the interrupt status word are set by conditions occurring after a convert-analog-input, convert-analog-input-with-external-synchronization, or read-and-convert-analog-input command is completed, but while the device is still busy. Since these errors cannot be indicated to the program by the condition code, interruptions result from setting on any ISW bit except the busy bit. Setting on any one of bits 3 through 7 also sets on the summary status indicator when the resulting interruption request occurs. After the interruption is accepted, executing any command other than read DSW resets the ISW.

The significant bits in the ISW and their meanings are:

Significant Bits	*Meaning*
3	Analog-to-digital converter inoperative.
5	Invalid analog data. The polarity of the input voltage changed during the conversion cycle.
7	Overload. The voltage read by the ADC was greater than the input voltage range selected by the amplifier.
12	Device busy. The analog input module is converting an input voltage.
13	Device end. The conversion operation requested is complete. If error conditions occur during the execution of I/O commands, this bit is set on along with other error identifying bits.
15	Invalid multiplexer address. An input point was addressed that is not installed.

Read DSW



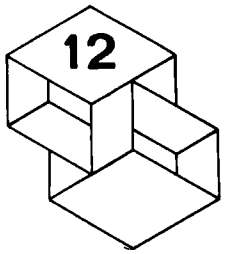
This command stores the 16-bit device status word (DSW) associated with the analog input module into the index register (R), or the accumulator if R=000. Execution of this command resets the bits in the DSW.

Device Status Word (DSW): The 16 bits in the device status word are set as a result of detected error conditions occurring during the execution of an immediate command to the I/O module. The operating program is notified of these error conditions by returning condition code 1 to the processor. If other commands are attempted before resetting the DSW bits, condition code 1 continues to be returned to the program.

The significant bits in the DSW and their meanings are:

Significant Bits	*Meaning*
1	Command reject. The analog input module cannot execute the command. (For example, an interruption-causing command issued to the device when it is disabled for interruptions.)
14	Data check. The analog input module has detected a parity error involving data. The operation that caused this condition usually can be retried successfully if the error condition is intermittent.

CHAPTER CONTENTS



The 5014 Analog Input Module Model D1 converts analog signals into binary values of 14 bits plus sign. Reed relays with a common-mode voltage up to 100V, or mercury relays with a common-mode voltage up to 250V can be used.

The model D1 analog input module can be located in any or all of the I/O module positions within an IBM 5026 enclosure. (Location 00 in the 5026 is the processor module location.) The 5014 Model E1 provides the capacity to expand a Model D1 an additional 128 points, in eight groups of 16 points each. Two model E1 modules may be attached to each model D1. The first model E1 module must be located directly below the model D1; the second model E1 must be directly below the first. The model E1 is addressed through the model D1.

FUNCTIONAL DESCRIPTION

In addition to logic controls, the model D1 I/O module consists of an analog-to-digital converter (ADC), an amplifier, and a point-select relay multiplexer capable of a maximum scanning speed of 100 points per second. As many as 128 two-wire differential analog input points can be accommodated, in eight groups of 16 each.

One amplifier is required to service all of the analog input points. If mercury relays are used and all analog inputs are at the ±5.12V level, the high-level amplifier is used. If a range of voltage levels must be accommodated, a multirange amplifier is used.

The multirange amplifier must be program controlled to one of seven different ranges. Analog signals to be converted can be within the range of −5.12V to +5.12V when using mercury relays. When using reed relays, the range is -0.64V to +5.12V. (See Appendix E for conversion of an analog voltage to its binary equivalent for the amplifier range selected.)

The analog-to-digital converter furnishes an overload indication in the data word when the signal level falls outside of a range that can be converted.

A temperature reference attachment is available for use in the cold junction compensation of thermocouple inputs.

I/O COMMANDS

The 5014 Analog Input Module Model D1 is programmed using the prepare I/O and halt I/O commands as described in Chapter 4 under "Input/Output Instruction." Immediate read and immediate write commands transfer data between the analog input module and the processor module. The setting of the modifier field bits in the command further defines the operation to be performed.

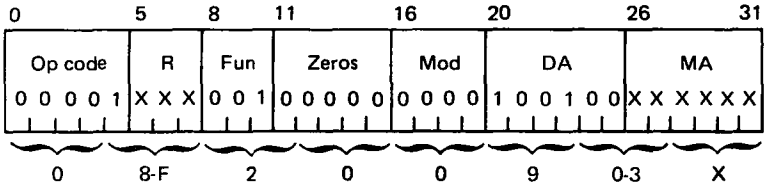
In all I/O commands addressed to the analog input module, the device address field is not used. The module address varies depending upon the physical location of the I/O module in the 5026 Enclosure.

Usually, a complete operation is performed within the time limit required for execution of the immediate commands. Exceptions to this rule are two convert commands that start the conversion cycle and then permit the program to resume normal operation while awaiting the interruption request presented by the module at the end of the conversion cycle.

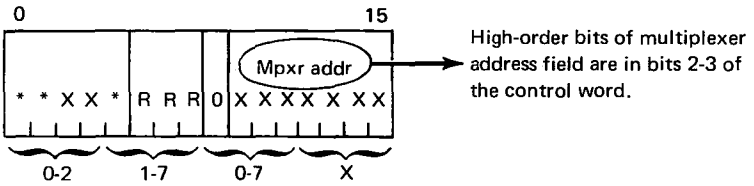
Executing either of the two convert commands sets on the ISW device-busy bit, which remains on during the conversion time. When the conversion is complete, an interruption request is presented to the processor, the device-busy bit is turned off, and the device-end bit is turned on. Condition code 2 is returned to all immediate commands issued (except read ISW or read DSW) while the module is busy or has a pending interruption.

The normal analog input programming routine is shown in Figure F-1 of "Appendix F."

Convert Analog Input



This command starts an analog-to-digital conversion cycle based upon control information contained in the index register (R), or the accumulator if R=000. The control information word has the format:



The range bits (RRR) in the control word specify the input voltage range and the amplifier gain desired, as shown in the following table.

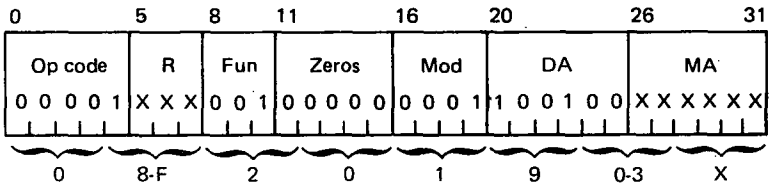
RRR Bits	Input Voltage Range	Amplifier Gain
000	Invalid	
001	±10 mV	512
010	±20 mV	256
011	±40 mV	128
100	±80 mV	64
101	±160 mV	32
110	±640 mV	8
111	±5.12V	1

The range bits (RRR) have no significance if the high-level amplifier is installed.

This convert-analog-input command starts an analog-to-digital conversion cycle of the analog input point specified by the multiplexer address field in the control word. An interruption request occurs at the end of the conversion cycle. The results of the conversion are then obtained by issuing a read-ADC or read-ADC-extended-precision command. The data format is in standard form (14 bits plus sign).

The multiplexer address field in the control word consists of bits 2-3 (the field high-order bits) and 9-15. The nine bits can be considered as a binary field to encode addresses 000 to 383. Bit 8 is reserved and set to zero. Bits 2-3 are necessary because an analog input configuration (models D1/E1/E1) can have up to 384 points.

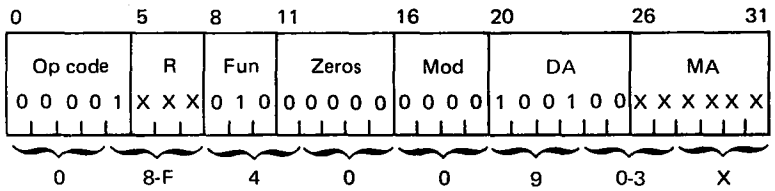
Convert Analog Input with External Synchronization



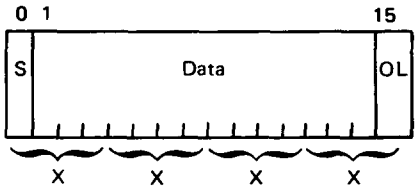
This command functions in the same manner as the convert-analog-input command except that the start of conversion depends upon the receipt of an external synchronization input pulse.

The user provides the external synchronization input pulse to start the analog-to-digital data conversion. (Refer to the *System/7 Installation Manual–Physical Planning,* Order No. GA34-0004, for more detailed data on the external synchronization input pulse.)

Read ADC

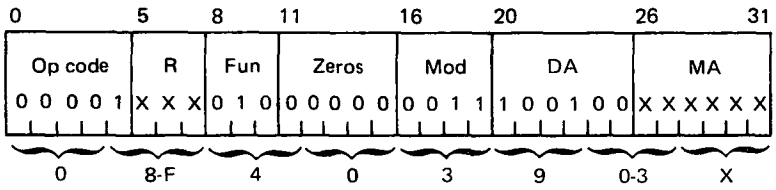


This command stores the binary output value from the ADC into the index register (R), or the accumulator if R=000. The 16 bits of register data have the format:



Fourteen bits of binary data plus sign (S bit) are returned. The single-bit OL field is set on if an overload condition occurred with the program-selected amplifier gain.

Read ISW



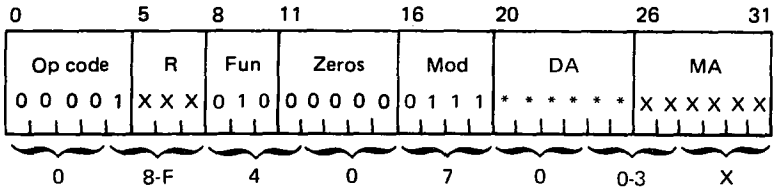
This command stores the 16-bit interrupt status word (ISW) associated with the analog input module into the index register (R), or the accumulator if R=000.

Interrupt Status Word (ISW): The 16 bits in the interrupt status word are set by conditions occurring after a convert-analog-input or a convert-analog-input-with-external-synchronization command is completed but while the device is still busy. Since these errors cannot be indicated to the program by the condition code, interruptions result from setting on any ISW bit except the busy bit. Setting on bits 3 or 7 also sets on the summary status indicator when the resulting interruption request occurs. After the interruption is accepted, executing any command other than read DSW resets the ISW.

The significant bits in the ISW and their meanings are:

Significant Bits	*Meaning*
3	Analog-to-digital converter inoperative.
7	Overload. The voltage read by the ADC was greater than the input voltage range selected by the amplifier.
8	Auto range selected in error.
12	Device busy. The analog input module is converting an input voltage.
13	Device end. The conversion operation requested is complete. If error conditions occur during the execution of I/O commands, this bit is set on along with other error identifying bits.
15	Invalid multiplexer address. An input point was addressed that was not installed.

Read DSW



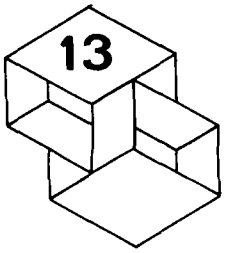
This command stores the 16-bit device status word (DSW) associated with the analog input module into the index register (R), or the accumulator if R=000. Execution of this command resets the bits in the DSW.

Device Status Word (DSW): The 16 bits in the device status word are set as a result of detected error conditions occurring during the execution of an immediate command to the I/O module. The operating program is notified of these error conditions by returning DSW bits, condition code 1 continues to be returned to the program.

The significant bits in the DSW and their meanings are:

Significant Bits	*Meaning*
1	Command reject. The analog input module cannot execute the command. (For example, an interruption-causing command issued to the device when it is disabled for interruptions.)
14	Data check. The analog input module has detected a parity error involving data. The operation that caused this condition usually can be retried successfully if the error condition is intermittent.

CHAPTER CONTENTS



The 5012 Multifunction Module Model A1 groups the following functions within one physical enclosure:

- As many as 32 analog input points
- As many as 128 digital input points
- As many as 64 digital output points
- One or two analog output points
- One 2790 Control for an IBM 2790 Data Communication System

Any combination of the above functions can be installed in a single multifunction module. Programming support, however, is provided for only four 2790 Controls in a System/7 configuration (this includes any 2790 Controls in 5013 Digital I/O Modules). With the single exception that *all* 2790 Controls must be located within the *same* 5026 Enclosure, the multifunction modules can be located in any or all of the I/O module positions within a 5026 Enclosure (location 00 in the 5026 is the processor module location).

FUNCTIONAL DESCRIPTION

Analog Input Control

An analog input control model B or model C can be used in the multifunction module. Features of these two models correspond to the features of the IBM 5014 Analog Input Module Models B1 and C1 except for the quantity of input points.

The analog input control model B consists of an analog-to-digital converter, an amplifier, and a point-select relay multiplexer capable of a maximum scanning speed of 200 points per second.

As many as 32 two-wire differential analog input points can be accommodated, in eight groups of four points each.

The analog input control model C consists of an analog-to-digital converter, an amplifier, and a differential solid-state multiplexer.

The solid-state multiplexer can have as many as 32 two-wire differential analog input points, in eight groups of four points each. With high-level inputs (±5.12V), a scanning speed of 20,000 points per second can be attained. A scanning speed of 14,000 points per second can be achieved with low-level inputs (10 to 640 mV). Use of automatic gain selection reduces the scanning speed to a maximum of 7,000 points per second.

Digital Input

As many as 128 isolated and/or non-isolated digital input points can be accommodated, in eight groups of 16 points each. Isolated two-terminal inputs can be activated by either contact or voltage sources provided by the user's equipment.

The first two digital input groups can each be provided with a special feature that permits them to interrupt the System/7 processor. A 16-point group is compared with a 16-bit reference register, and interruptions can be initiated on the basis of either an equal or an unequal comparison.

A non-isolated digital input group provides 16 latching or non-latching points of contact sensing capability. A non-isolated group cannot, however, have the process interrupt feature.

Digital Output

As many as 64 isolated and/or non-isolated digital output points can be accommodated, in four groups of 16 points each. Each isolated group can use either of the two optional digital output circuits available (medium-power group or contact group). These circuits are described in more detail in the *System/7 Installation Manual–Physical Planning,* Order No. GA34-0004.

A medium-power non-isolated digital output group provides 16 solid-state switches for switching up to 52.8V dc and 450 ma. of user-supplied power.

Analog Output

One or two isolated analog output points can be accommodated within the multifunction module. Each point provides a voltage output to user terminals.

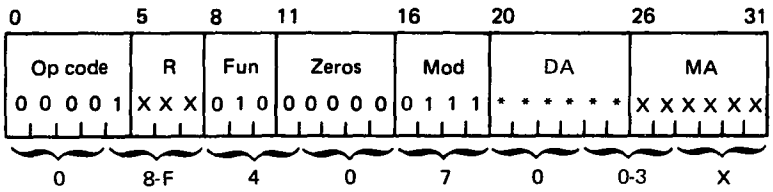
2790 Control

A single loop of the IBM 2790 Data Communication System can be attached to the multifunction module. Such an attachment allows the System/7 to act as system controller to the 2790 devices (2791/2793 Area Stations, 2792 Remote Communications Controller, and 2795/2796/2797 Data Entry Units). Data enters the 2790 system from a card reader, badge reader, pulse counter, or key entry unit. Output data to the 2790 system is sent to 2791 indicator lights or to a printer attached to either a 2791 or 2793 Area Station.

COMMON I/O COMMANDS

Three I/O commands are common to all functions in the multifunction module: read DSW, read ID, and read-ID-extension. In these three commands, the contents of the device address field are ignored, but the module address is determined by the physical location of the multifunction module in the 5026 Enclosure. The read ID and read-ID-extension commands are described earlier in Chapter 4 under "Input/Output Instruction," since these two commands apply to all I/O modules.

Read DSW



The read DSW command stores the 16-bit device status word (DSW) associated with the multifunction module into the index register (R), or the accumulator if R=000. Execution of this command resets the bits in the DSW.

Device Status Word (DSW): The 16 bits in the device status word are set as a result of detected error conditions occurring during the execution of an immediate command to the I/O module. The operating program is notified of these error conditions by returning condition code 1 to the processor. If other commands are attempted before resetting the DSW bits, condition code 1 continues to be returned to the program.

The significant bits in the DSW and their meanings are:

Significant Bits	*Meaning*
1	Command reject. The multifunction module cannot execute the command. (For example, an interruption-causing command issued to the device when it is disabled for interruptions.)
3	Open circuit. A digital input circuit breaker is open when the digital input subaddress was selected.
14	Data check. The multifunction module has detected a parity error involving data. The operation that caused this condition usually can be retried successfully if the error condition is intermittent.
15	Invalid device address. An immediate command was issued to a device that is not installed in the multifunction module.

MODULE DEVICE ADDRESSES

I/O commands to the multifunction module use the six-bit device address field (bits 20 to 25 of the PIO instruction) to specify the particular function being addressed. The device address bit assignments for each function are:

Device Address	*Function*
000000	Digital input group 0 with or without process interrupt feature
000100	Digital input group 1 with or without process interrupt feature
001000	Digital input group 2
001100	Digital input group 3
010000	Digital input group 4
010100	Digital input group 5
011000	Digital input group 6
011100	Digital input group 7
100000	2790 Control
100100	Analog input
101000	Analog output point 0
101100	Analog output point 1
110000	Digital output group 0
110100	Digital output group 1
111000	Digital output group 2
111100	Digital output group 3

The prepare I/O command is used as described in Chapter 4 under "Input/Output Instruction." Only four device addresses are used with the prepare I/O command in order to prepare an input source to interrupt. The four device addresses and their functions are:

Device Address	*Function*
000000	Prepare digital input group 0 with process interrupt feature
000100	Prepare digital input group 1 with process interrupt feature
100000	Prepare 2790 Control
100100	Prepare analog input

ANALOG INPUT CONTROL MODEL B

In the multifunction module, analog input control model B and model C are mutually exclusive.

The analog input control model B converts analog signals into binary values of 14 bits plus sign. These analog signals can be within the range of ±10 mV full scale to ±5.12V full scale.

The analog input function is accomplished by an analog-to-digital converter (ADC), an amplifier, and a point-select relay multiplexer capable of a maximum scanning speed of 200 points per second.

As many as 32 two-wire differential analog input points can be accommodated, in eight groups of four points each.

One amplifier is required to service all of the analog input points. If all analog inputs are at the ±5.12V level, the high-level-only amplifier is used. If a range of voltage levels must be accommodated, a multirange amplifier is used. The multirange amplifier can be program controlled to one of the seven different ranges, or to automatically select the appropriate amplifier gain. When the amplifier is programmed to automatically select the gain, analog signal resolution is reduced to 12 bits plus sign if the selected gain is returned (binary encoded in three bits) to the program with the data. The multirange amplifier allows conversion of signals in the ±10 mV, ±20 mV, ±40 mV, ±80 mV, ±160 mV, ±640 mV, and ±5.12V ranges. (See Appendix E for conversion of an analog voltage to its binary equivalent for the amplifier range selected.)

The analog-to-digital converter furnishes an overload indication in the data word when the signal level falls outside of a range that can be converted.

A temperature reference attachment is available for use in the cold junction compensation of thermocouple inputs.

I/O Commands

The prepare I/O and halt I/O commands are used as described in Chapter 4 under "Input/Output Instruction" and in this chapter under "Module Device Addresses."

Immediate read and immediate write commands transfer data between the analog input control and the processor module. The setting of the modifier field bits in the command further defines the operation to be performed.

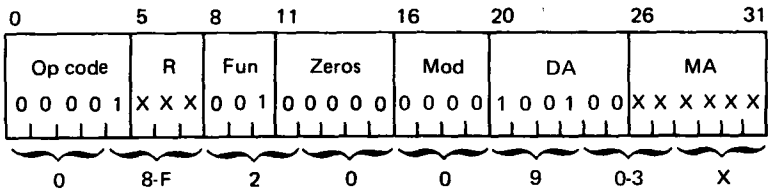
In all I/O commands addressed to analog input, the device address field contents must be 100100. The module address varies depending upon the physical location of the multi-function module in the 5026 Enclosure.

Usually, a complete operation is performed within the time limit required for execution of the immediate commands. Exceptions to this rule are two convert commands that start the conversion cycle and then permit the program to resume normal operation while awaiting the interruption request presented by the module at the end of the conversion cycle.

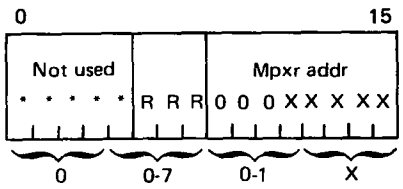
Executing either of the two commands sets on the ISW device-busy bit, which remains on during the conversion time. When the conversion is complete, an interruption request is presented to the processor, the device-busy bit is turned off, and the device-end bit is turned on. Condition code 2 is returned to all immediate commands issued to this device address (except read ISW) while the module is busy or has a pending interruption.

The normal analog input programming routine is shown in Figure F-1 of "Appendix F."

Convert Analog Input



This command starts an analog-to-digital conversion cycle based upon control information contained in the index register (R), or the accumulator if R=000, to the analog input module. The control information word has the format:



The range bits (RRR) in the control word specify the input voltage range and the amplifier gain desired, as shown in the following table.

RRR Bits	*Input Voltage Range*	*Amplifier Gain*
000	–	Automatic gain
001	±10 mV	512
010	±20 mV	256
011	±40 mV	128
100	±80 mV	64
101	±160 mV	32
110	±640 mV	8
111	±5.12V	1

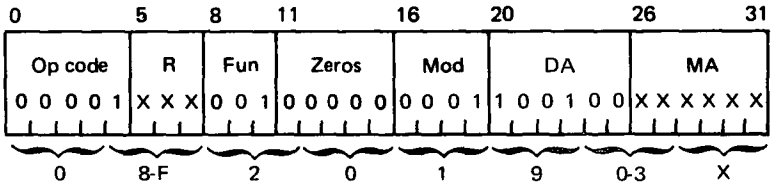
When automatic gain is specified (RRR bits=000), the optimum gain (one of the seven amplifier gains) is selected automatically, based upon the input signal voltage. The selected gain can be returned to the program, along with the converted signal data, by issuing a read-ADC command. Since the indication of gain requires three bits, the input data itself is reduced to 12 bits plus sign.

The range bits (RRR) have no significance if the high-level amplifier is installed.

This convert-analog-input command starts an analog-to-digital conversion cycle of the analog input point specified by the multiplexer address field in the control word. An interruption request occurs at the end of the conversion cycle. The results of the conversion are then obtained by issuing a read-ADC or read-ADC-extended-precision command.

With the read-ADC command, the data format is in short form (12 bits plus sign) if automatic gain was specified, or in standard form (14 bits plus sign) if a programmed range was specified. The read-ADC-extended-precision command always returns the data in standard form.

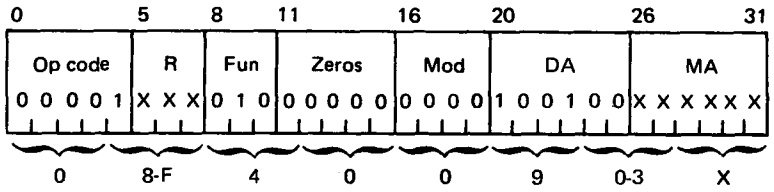
Convert Analog Input with External Synchronization



This command functions in the same manner as the convert-analog-input command except that the start of conversion depends upon the receipt of an external synchronization input pulse.

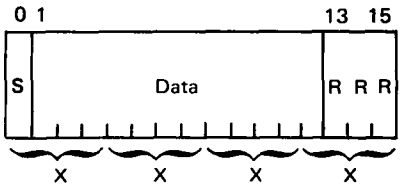
The user provides the external synchronization input pulse to start the analog-to-digital data conversion. (Refer to the *System/7 Installation Manual–Physical Planning*, Order No. GA34-0004, for more detailed data on the external synchronization input pulse.)

Read ADC



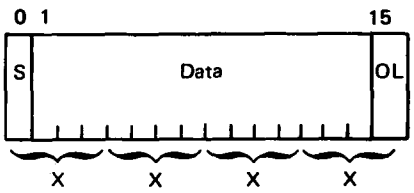
This command stores the binary output value from the ADC into the index register (R), or the accumulator if R=000.

Data stored in the register can have either of two formats, depending upon whether the last convert command requested automatic gain or selected a particular amplifier gain. If automatic gain was requested, the 16 bits of data have the format:



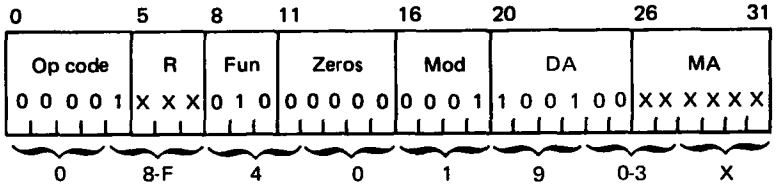
Twelve bits of binary data plus sign (S bit) are returned. The RRR field indicates which amplifier gain was automatically selected by the amplifier. (Refer to the table in the "Convert Analog Input" command description.) If RRR=000, an overload condition occurred with the automatic gain selection.

If a particular amplifier gain was selected by the last convert command, the 16 bits of data have the format:



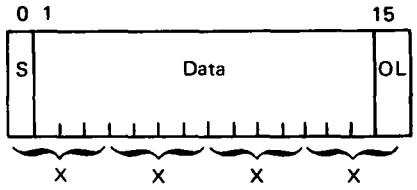
Fourteen bits of binary data plus sign (S bit) are returned. The single-bit OL field is set on if an overload condition occurred with the program-selected amplifier gain.

Read ADC Extended Precision



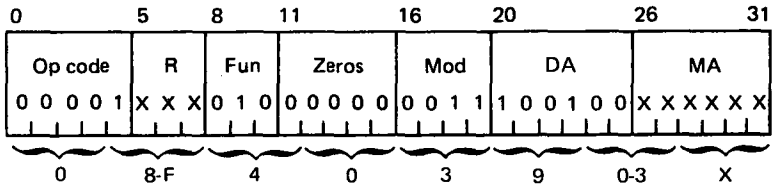
This command stores in standard form (14 bits plus sign) the binary output value from the ADC when automatic amplifier gain was selected by the last convert command.

The binary output value resulting from the last conversion cycle is stored in the index register (R), or the accumulator if R=000. The data transferred to the register has the format:



Fourteen bits of binary data plus sign (S bit) are returned. The single-bit OL field is set on if an overload condition occurred with the automatic amplifier gain selection. Note that the automatically selected gain is not returned with the data. This gain can be determined by issuing a read-ADC command before or after the read-ADC-extended-precision command.

Read ISW



This command stores the 16-bit interrupt status word (ISW) associated with the analog input control into the index register (R), or the accumulator if R=000.

Interrupt Status Word (ISW): The 16 bits in the interrupt status word are set by conditions occurring after a convert-analog-input or convert-analog-input-with-external-synchronization command is completed but while the device is still busy. Since these errors cannot be indicated to the program by the condition code, interruptions result from setting on any ISW bit except the busy bit. Setting on any one of bits 3 through 7 also sets on the summary status indicator when the resulting interruption request occurs. After the interruption is accepted, the execution of any command directed to the analog input control resets the ISW.

The significant bits in the ISW and their meanings are:

Significant Bits	*Meaning*
3	Analog-to-digital converter inoperative.
5	Multiple relays selected. This multiplexer failure is caused by more than one input being selected during the conversion cycle.
6	No relays selected. This multiplexer failure results from no input being selected during the conversion cycle.
7	Overload. The voltage read by the ADC was greater than the input voltage range selected by the amplifier.
12	Device busy. The analog input control is converting an input voltage.
13	Device end. The conversion operation requested is complete. If error conditions occur during the execution of such operations as analog-to-digital conversions, this bit is set on along with other error identifying bits.
15	Invalid multiplexer address. An input point was addressed that was not installed.

ANALOG INPUT CONTROL MODEL C

The analog input control model C converts analog signals into binary values of 14 bits plus sign. These analog signals can be within the range of ±10 mV full scale to ±5.12V full scale. In the multifunction module, analog input control model B and model C are mutually exclusive.

The analog input function is accomplished by an analog-to-digital converter (ADC), an amplifier, and a differential solid-state multiplexer.

The solid-state multiplexer can have as many as 32 two-wire differential analog input points, in eight groups of four points each. With high-level inputs (±5.12V), a speed of 20,000 points per second can be attained. An input speed of 14,000 points per second can be achieved with low-level inputs (10 to 640 mV). A scanning speed of 7,000 points per second can be obtained with the use of automatic gain selection. The actual scanning rate obtained, however, depends on the input voltage value, the physical location of the I/O module, and the programming techniques used. Conversion time required by the ADC varies from 25 to 46 microseconds, depending on the input voltage. Additional delays due to module location and programming must also be considered when computing the actual scanning rate.

One amplifier is required to service all of the analog input points. If all analog inputs are at the ±5.12V level, the high-level-only amplifier is used. If a range of voltage levels must be accommodated, a multirange amplifier is used. The multirange amplifier can be program controlled to one of the seven different ranges, or to automatically select the appropriate amplifier gain. When the amplifier is programmed to automatically select the gain, analog signal resolution is reduced to 12 bits plus sign if the selected gain is returned (binary encoded in three bits) to the program with the data. The multirange amplifier allows conversion of signals in the ±10 mV, ±20 mV, ±40 mV, ±80 mV, ±160 mV, ±640 mV, and ±5.12 V ranges. (See Appendix E for conversion of an analog voltage to its binary equivalent for the amplifier range selected.)

The analog-to-digital converter furnishes an overload indication in the data word when the signal level falls outside of a range that can be converted.

A temperature reference attachment is available for use in the cold junction compensation of thermocouple inputs.

I/O Commands

The prepare I/O and halt I/O commands are used as described in Chapter 4 under "Input/Output Instruction" and in this chapter under "Module Device Addresses."

Immediate read and immediate write commands transfer data between the analog input control and the processor module. The setting of the modifier field bits in the command further defines the operation to be performed.

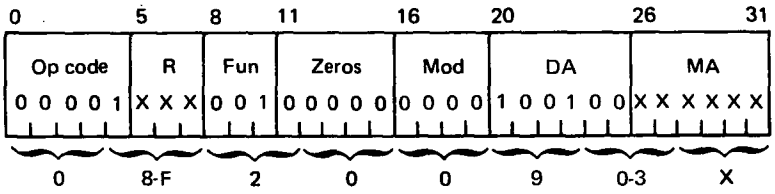
In all I/O commands addressed to analog input, the device address field contents must be 100100. The module address varies depending upon the physical location of the multifunction module in the 5026 Enclosure.

Usually, a complete operation is performed within the time limit required for execution of the immediate commands. Exceptions to this rule are three convert commands that start the conversion cycle and then permit the program to resume normal operation while awaiting the interruption request presented by the module at the end of the conversion cycle.

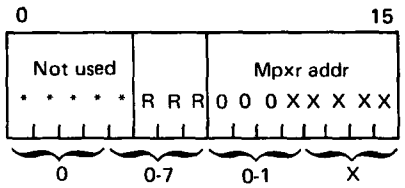
Executing any of the three commands sets on the ISW device-busy bit, which remains on during the conversion time. When the conversion is complete, an interruption request is presented to the processor, the device-busy bit is turned off, and the device-end bit is turned on. Condition code 2 is returned to all immediate commands issued to this device address (except read ISW) while the module is busy or has a pending interruption.

The normal analog input programming routine is shown in Figure F-1 of "Appendix F."

Convert Analog Input



This command starts an analog-to-digital conversion cycle based upon control information contained in the index register (R), or the accumulator if R=000. The control information word has the format:



The range bits (RRR) in the control word specify the input voltage range and the amplifier gain desired, as shown in the following table.

RRR Bits ·	*Input Voltage Range*	*Amplifier Gain*
000	–	Automatic gain
001	±10 mV	512
010	±20 mV	256
011	±40 mV	128
100	±80 mV	64
101	±160 mV	32
110	±640 mV	8
111	±5.12V	1

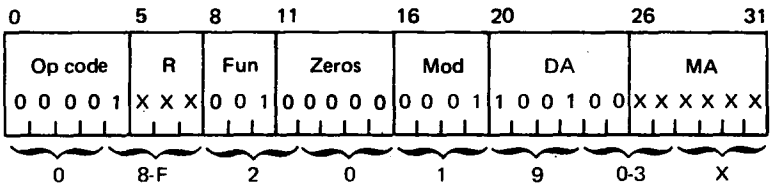
When automatic gain is specified (RRR bits=000), the optimum gain (one of the seven amplifier gains) is selected automatically, based upon the input signal voltage. The selected gain can be returned to the program, along with the converted signal data, by issuing a read-ADC command. Since the indication of gain requires three bits, the input data itself is reduced to 12 bits plus sign.

The range bits (RRR) have no significance if the high-level amplifier is installed.

This convert-analog-input command starts an analog-to-digital conversion cycle of the analog input point specified by the multiplexer address field in the control word. An interruption request occurs at the end of the conversion cycle. The results of the conversion are then obtained by issuing a read-ADC, a read-ADC-extended-precision or a read-and-convert-ADC command.

With either the read-ADC command or the read-and-convert-ADC command, the data format is in short form (12 bits plus sign) if automatic gain was specified, or in standard form (14 bits plus sign) if a particular range was specified. The read-ADC-extended-precision command always returns the data in standard form.

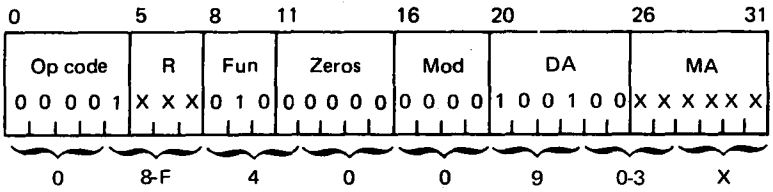
Convert Analog Input with External Synchronization



This command functions in the same manner as the convert-analog-input command except that the start of conversion depends upon the receipt of an external synchronization input pulse.

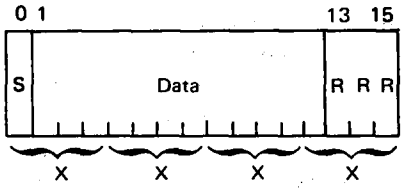
The user provides the external synchronization input pulse to start the analog-to-digital data conversion. (*System/7 Installation Manual–Physical Planning,* Order No. GA34-0004, contains more detailed data on the external synchronization input pulse.)

Read ADC



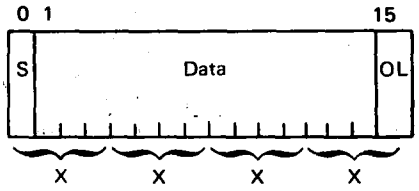
This command stores the binary output value from the ADC into the index register (R), or the accumulator if R=000.

Data stored in the register can have either of two formats, depending upon whether the last convert command requested automatic gain or selected a particular amplifier gain. If automatic gain was requested, the 16 bits of data have the format:



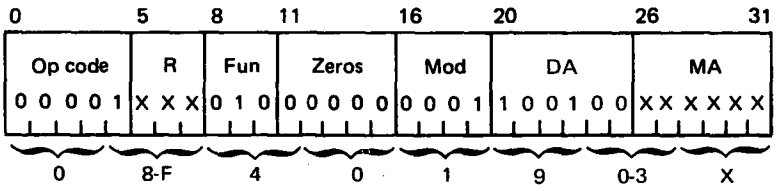
Twelve bits of binary data plus sign (S bit) are returned. The RRR field indicates which amplifier gain was automatically selected by the amplifier. (Refer to the table in the "Convert Analog Input" command description.) If RRR=000, an overload condition occurred with the automatic gain selection.

If a particular amplifier gain was selected by the last convert command, the 16 bits of data have the format:



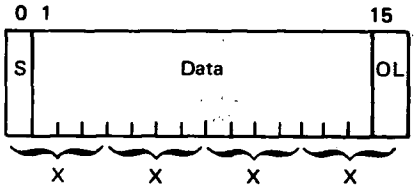
Fourteen bits of binary data plus sign (S bit) are returned. The single-bit OL field is set on if an overload condition occurred with the program-selected amplifier gain.

Read ADC Extended Precision



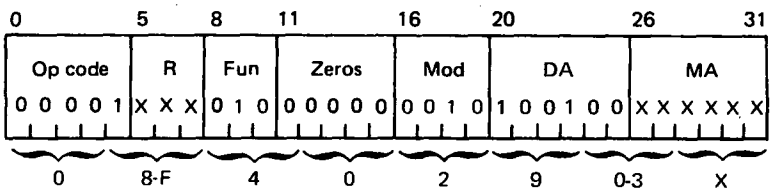
This command stores in standard form (14 bits plus sign) the binary output value from the ADC, when automatic amplifier gain was selected by the last convert command.

The binary output value resulting from the last conversion cycle is stored in the index register (R), or the accumulator if R=000. The data transferred to the register has the format:



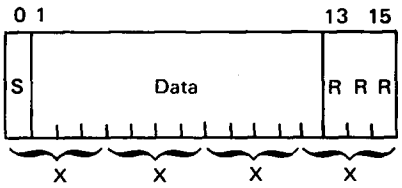
Fourteen bits of binary data plus sign (S bit) are returned. The single-bit OL field is set on if an overload condition occurred with the automatic amplifier gain selection. Note that the automatically selected gain is not returned with the data. This gain can be determined by issuing a read-ADC command before or after the read-ADC-extended-precision command.

Read and Convert ADC



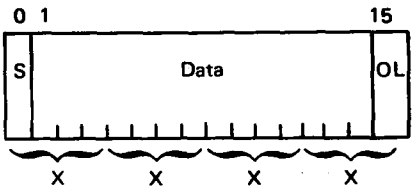
This command is used for repetitive operations on a single analog input point. The read-and-convert-ADC command stores the binary output value from the ADC into the index register (R), or the accumulator if R=000, and starts another conversion cycle.

Data transferred to the register can have either of two formats, depending upon whether the last convert command requested automatic gain or selected a particular amplifier gain. If automatic gain was selected, the 16 bits of data have the format:



Twelve bits of binary data plus sign (S bit) are returned. The RRR field indicates which amplifier gain was automatically selected by the amplifier. (Refer to the table in the "Convert Analog Input" command description.) If RRR=000, an overload condition occurred with the automatic gain selection.

If a particular amplifier gain was selected by the last convert command, the 16 bits of data have the format:

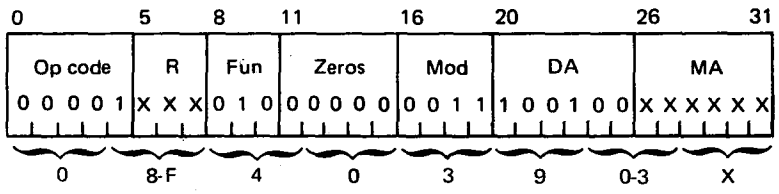


Fourteen bits of binary data plus sign (S bit) are returned. The single-bit OL field is set on if an overload condition occurred with the program-selected amplifier gain.

The convert portion of the read-and-convert-ADC command converts the analog input point specified by the last convert command, utilizing the same amplifier gain. Thus, if automatic gain was specified by the last convert command, then automatic gain is also used by each subsequent execution of the read-and-convert-ADC command.

Since this command performs repetitive operations on a single analog input point, the solid-state multiplexer is monopolized during this time. To terminate this repetitive read sequence and free the multiplexer, a read-ADC command must be issued if the result of the last conversion cycle is needed by the program. However, any command other than the read-and-convert-ADC command may be issued to terminate the repetitive read sequence.

Read ISW



This command stores the 16-bit interrupt status word (ISW) associated with the analog input control into the index register (R), or the accumulator if R=000.

Interrupt Status Word (ISW): The 16 bits in the interrupt status word are set by conditions occurring after a convert-analog-input, convert-analog-input-with-external-synchronization, or read-and-convert-analog-input command is completed, but while the device is still busy. Since these errors cannot be indicated to the program by the condition code, interruptions result from setting on any ISW bit except the busy bit. Setting on any one of bits 3 through 7 also sets on the summary status indicator when the resulting interruption request occurs. After the interruption is accepted, the execution of any command directed to the analog input control resets the ISW.

The significant bits in the ISW and their meanings are:

Significant Bits	*Meaning*
3	Analog-to-digital converter inoperative.
5	Invalid analog data. The polarity of the input voltage changed during the conversion cycle.
7	Overload. The voltage read by the ADC was greater than the input voltage range selected by the amplifier.

Significant Bits	*Meaning*
12	Device busy. The analog input control is converting an input voltage.
13	Device end. The conversion operation requested is complete. If error conditions occur during the execution of such operations as analog-to-digital conversions, this bit is set on along with other error identifying bits.
15	Invalid multiplexer address. An input point was addressed that was not installed.

DIGITAL INPUT CONTROL

The digital input control function of the multifunction module can accommodate as many as 128 isolated and/or non-isolated digital input points, in eight groups of 16 points each. These points are two-terminal inputs that can be activated by either contact or voltage sources provided by the user's equipment.A non-isolated digital input group provides 16 latching or non-latching points of contact sensing capability.

A digital input contact point has a 1 value in its corresponding bit when the contact is closed, and a 0 value when the contact is open.

An isolated digital input voltage point has a 1 value in its corresponding bit when the input is from +2V to +52.8V, and a 0 value when the input is from −52.8V to +0.6V. A non-isolated digital input voltage point has the opposite polarity sensing; a 0 value from +2V to +52.8V, and a 1 value from −52.8V to +0.6V. An input between +0.6V and +2V, in either case, produces an indeterminate result.

Each group of digital input points can be program controlled to provide either latched or unlatched input. In the latched mode of operation, each bit that is set on in the group remains in that state (regardless of any further change in input signal or contact operation) until the group is reset or unlatched. A bit is not reset if the input signal remains active (voltage present or switch closed) throughout the execution of the read/reset digital input register command. In the unlatched mode of operation, each bit in the group is set on or off dynamically as determined by the input fluctuations.

Digital input circuits are compatible with digital output circuits, which permits multiplexing of digital output points with the user's contacts for digital input sensing applications. If all of the points are isolated, either positive or negative voltage sources can be used through proper connection of input sources to the input terminals. This circuit compatibility also permits direct connection of digital input points to digital output points for testing of the input/output circuits.

Process Interrupt Feature

A Process Interrupt feature can be obtained for one or both of the first two digital input groups (group 0 and/or group 1). Each of the two groups is a separate interrupting source and, therefore, can be assigned its own priority level and sublevel by separate prepare I/O commands. Group 0 is installed when only one group is ordered. The Process Interrupt feature can be attached only to isolated groups.

With the Process Interrupt feature, a 16-point group is compared with a 16-bit reference register (which can be set to any value by the program), and an interruption request is presented on the basis of either an equal or an unequal bit-by-bit comparison. The choice of comparison is under program control and can be changed at any time. When a group has been conditioned to perform a compare operation, the program cannot read the digital input register for that group until an interruption request is presented. After the request is presented, the group will no longer be conditioned to perform a compare operation. The group can then be used the same as a noninterrupting digital input group.

The Process Interrupt feature can be used with groups 0 and/or 1 when they are operating in either the latched or unlatched mode of operation. In the latched mode, each bit that is set on in the group remains in that state (regardless of any further change in input signal or contact operation) until the group is reset or unlatched. All bits in the group are reset, even if the input signal remains active throughout the execution of the read/reset digital input register command. In order to again set a bit in the digital input register, a positive transition must occur; for example, the input point must have gone to a 0 value (+0.6V to −52.8V) and then back to a 1 value (+2V to +52.8V) after the digital input register was reset. When changing modes, latched to unlatched or unlatched to latched, and the user's input value is in the logical 1 state, the digital input register indicates a logical 0 until such time as another positive transition takes place.

In the unlatched mode of operation, each bit in the group is set on or off dynamically as determined by the input signal fluctuations. The output of the digital input register follows the user's input value; for example, the user's value at a logical 0 results in a logical 0 output from the digital input register.

Exception: The output of the digital input register will *not* follow the user's input value in unlatched mode whenever a switch from latched mode to unlatched mode takes place and the user's input value remains in the logical 1 state. Either of two methods can be used to restore the digital input register to again reflect the user's input values.

1. Program the following sequence:
   —Issue a set-test-signal command (bit 15=0).
   —Establish a 3-microsecond delay.
   —Issue a read/reset-digital-input-register command.
2. Ensure that the user's input value returns to a logical 0 after the completion of latched mode operation.

Even though the Process Interrupt feature is installed, it can be disabled and enabled under program control at any time. Program control of the Process Interrupt feature is provided by the set-digital-input-interrupt-control command.

I/O Commands

The prepare I/O command is used as described in Chapter 4 under "Input/Output Instruction" and in this chapter under "Module Device Addresses." This command applies only to groups 0 and 1 because only these two groups can have the Process Interrupt feature. The prepare I/O device address field must be 000000 for group 0 or 000100 for group 1.

Data transfers between the digital input control and the processor module are accomplished by immediate read and immediate write commands. The setting of the modifier field bits in the command further defines the operation to be performed.

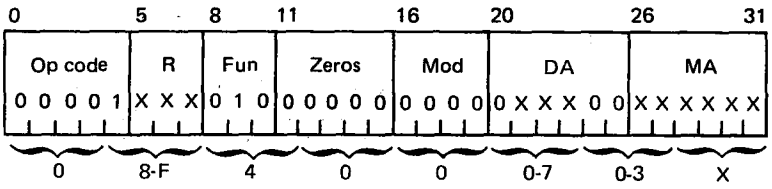
All read and write I/O commands addressed to the digital input control must use one of the following device addresses in order to address the corresponding digital input group separately:

Device Address	*Digital Input Group*
000000	0
000100	1
001000	2
001100	3
010000	4
010100	5
011000	6
011100	7

The module address in the I/O command varies depending upon the physical location of the multifunction module in the 5026 Enclosure.

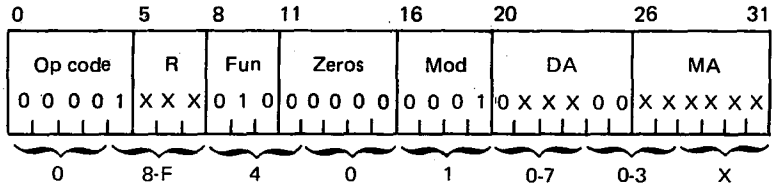
The normal programming routines for digital input are shown in Figures F-3 and F-4 of "Appendix F."

Read Digital Input Register



This command stores the 16 bits of input data from a group (identified by the device address field) into the index register (R), or the accumulator if R=000.

Read/Reset Digital Input Register



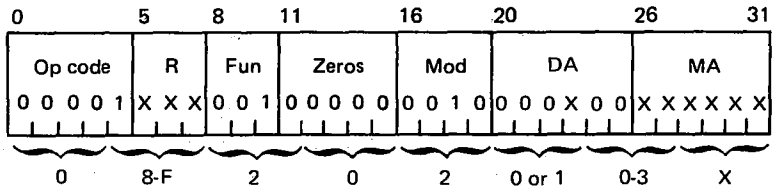
This command stores the 16 bits of input data from a group (identified by the device address field) into the index register (R), or the accumulator if R=000.

If a group is non-isolated and operating in latched mode, its digital input register is reset to 0 only if its input has returned to a 0 value. If the input is at a 1 level, the register position remains set to 1.

If the group is isolated, operating in latched mode, and the input is not active, its digital input register is then reset to 0.

If a process interrupt group is operating in latched mode, its digital input register is then reset to a 0 value (even if the input is still active).

Set Digital Input Reference Register

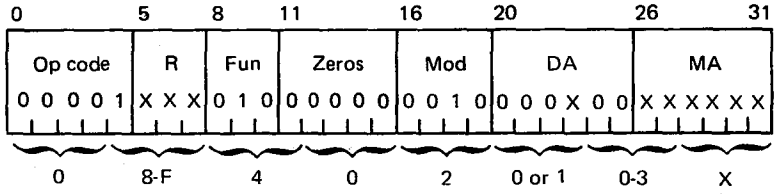


This command stores a 16-bit value in the reference register associated with group 0 or group 1. The choice is determined by the device address field in the command: 000000 for group 0 or 000100 for group 1.

The value set into the reference register is obtained from the index register (R), or the accumulator if R=000.

The reference registers are part of the Process Interrupt feature for groups 0 and 1, and are used for comparison with the 16-point group input register. Interruption requests can be presented for either an equal or an unequal comparison; the choice is under control of the program. (Refer to "Set Digital Input Interrupt Control" later in this chapter.)

Read Digital Input Reference Register

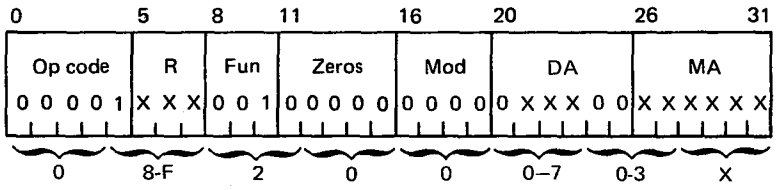


This command stores the contents of the 16-bit reference register associated with group 0 or group 1. The choice is determined by the device address field in the command: 000000 for group 0 or 000100 for group 1.

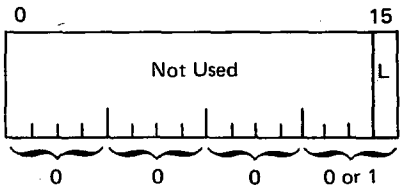
The reference register contents are stored into the index register (R), or the accumulator if R=000.

The reference registers are part of the Process Interrupt feature for groups 0 and 1, and are used for comparison with the 16-point group input register. Interruption requests can be presented for either an equal or an unequal comparison; the choice is under control of the program. (Refer to "Set Digital Input Interrupt Control" later in this chapter.)

Set Digital Input Latch Control



This command sends control information to the digital input group identified by the device address field. The control information is indicated by the value of bit 15 of the index register (R), or the accumulator if R=000, and has the following format:



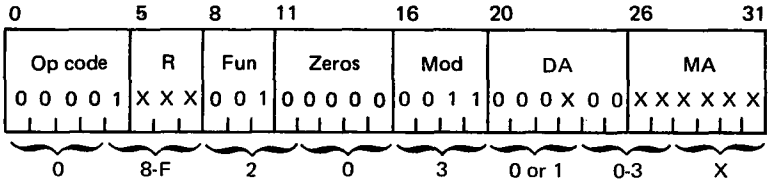
Control information in bit 15 can be directed to any of the eight possible digital input groups (groups 0 to 7) in a multifunction module. The setting of bit 15 determines whether the selected group is to operate in the latched or unlatched mode. Bit 15 set to a 1 value specifies the latched mode, and a 0 value specifies the unlatched mode.

When the Process Interrupt feature is installed, changing modes (latched to unlatched or unlatched to latched) resets the digital input register to the all 0's state. The register will remain all 0's until the next positive transition of the user's input signal (logical 0 to 1). Setting bit 15=0 also resets the digital input register to the all 0's state.

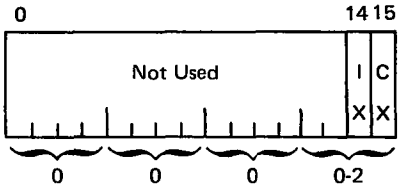
Note. If while in the latched mode of operation, the user's signal has remained in the logical 1 state when unlatched mode of operation is selected, the group must be reinitiated before normal unlatched mode operation takes place. Either of two methods can be used to restore the digital input register to again reflect the user's input values.

1. Program the following sequence:
   - –Issue a set-test-signal command (bit 15=0).
   - –Establish a 3-microsecond delay.
   - –Issue a read/reset-digital-input-register command.
2. Ensure that the user's input value returns to a logical 0 after the completion of latched mode operation.

Set Digital Input Interrupt Control



This command sends control information to the Process Interrupt feature of digital input group 0 or 1. The choice is determined by the device address field in the command: 000000 for group 0 or 000100 for group 1. The control information is indicated by the value of bits 14 and 15 of the index register (R), or the accumulator if R=000, and has the following format:

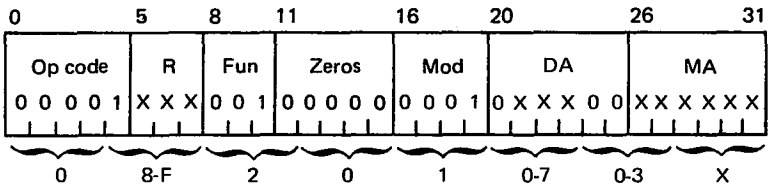


Forced interruption control is provided by the setting of bit 14. If bit 14 is on, a process interrupt is forced for the selected group (group 0 or 1). The setting of bit 15 determines if interruption requests are to be presented for an equal or an unequal comparison between the selected digital input group register and the reference register. When bit 15 is set on, an interruption request is presented to the system for an equal comparison. When bit 15 is set off, the interruption request occurs for an unequal comparison.

The set-digital-input-interrupt-control command turns on the device-busy bit in the digital input ISW. The addressed digital input group remains busy to all subsequent commands (except another set-digital-input-interrupt-control with bit 14 on or a read ISW command) until an interruption request has been presented by the group.

This command applies to digital input groups 0 and 1 only if they have the Process Interrupt feature installed. If this feature is not installed, the command does not apply to the Digital Input feature.

Set Test Signal



In unlatched mode of operation, this command sets a digital input register to a predetermined value, regardless of the user's input value. Selection of the desired digital input group is made by the device address field contents in the command to the addressed multifunction module.

The value to be set into the digital input register is determined by the setting of bit 15 in the index register (R), or the accumulator if R=000. If bit 15=0, the 16-bit digital input register is set to all 0's. If bit 15=1, the digital input register is set to all 1's. A 3-microsecond delay must precede the issuance of a read/reset or read-digital-input-register command to obtain the test data. A read/reset-digital-input-register command must be issued to restore the digital input circuits to their original status.

Digital Input Groups Without Process Interrupt Feature: In latched mode, operation is the same as in unlatched mode except under these two conditions.

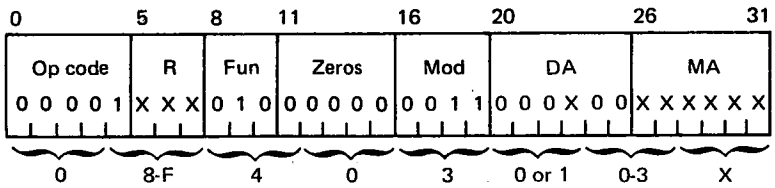
1. Whenever the following sequence of instructions is issued, two successive read/reset digital-input-register commands must be issued to restore the digital input circuits to their original status:
   –Set-digital-input-latch-control (bit 15=1).
   –Set-test-signal (bit 15=1).
   –3-microsecond delay.
2. The following sequence of instructions results in reading the user's input value instead of reading all 0's:
   –Set-digital-input-latch-control (bit 15=1).
   –Set-test-signal (bit 15=0).
   –3-microsecond delay.

Digital Input Groups With Process Interrupt Feature: In latched mode, operation is the same as unlatched mode except under the following conditions:

When any of the user's input signals are in the logical 1 state at the same time that a set-digital-input-latch-control command is issued (bit 15=1) followed by a set-test-signal-to-1's instruction, a read-digital-input-register command or a read/reset-digital-input-register command will read 1's for all bits whose inputs were at the logical 0-state.

Note. If the sequence of commands is such that a set-test-signal command (bit 15=0 or bit 15=1) is followed by a mode change (latched to unlatched or unlatched to latched), the digital input register is set to the all 0's state.

Read ISW



This command stores a 16-bit interrupt status word (ISW) into the index register (R), or the accumulator if R=000.

Interrupt Status Word (ISW): No ISW is associated with the digital input function unless the process interrupt feature is installed. For each process interrupt feature (maximum of two), there is a separate ISW, one for each group with interruption capability.

The setting of ISW bits (except the busy bit) is indicated to the operating program by an interruption request. Interruption requests are presented because of errors detected in digital input operations, as well as for the occurrence of a process interrupt.

A read ISW command also resets the addressed ISW. The significant bits in the ISW and their meanings are:

Significant Bits	*Meaning*
3	Open circuit. A digital input circuit breaker has opened during the time that the corresponding digital input group is busy. This condition results in an interruption request.
12	Device busy. Digital input group 0 or 1 was addressed and that group is in an interruption-enabled state. (See "Set Digital Input Interrupt Control" previously described in this chapter.) The resulting interruption on equal or unequal comparison turns off this bit and turns on the device-end bit.
13	Device-end. A successful compare on equal or unequal has occurred for group 0 or 1. This causes an interruption request and disables interruptions for that group.

DIGITAL OUTPUT CONTROL

The Digital Output Control feature of the multifunction module can accommodate as many as 64 isolated and/or non-isolated digital output points, in four groups of 16 points each. Each group can be configured to use one of the three optional digital output circuit types available (low-power group, medium-power group, and contact group). A non-isolated digital output group provides 16 medium-power, solid-state switches for switching up to 52.8V dc and 450 ma. of user-supplied power.

The digital output points can be operated by either of two methods: (1) data can be sent directly to a selected digital output group register which, in turn, operates the output points; or (2) data can be sent to the holding register and then read back and checked for accuracy before transferring it to one of four digital output group registers.

Digital output circuits are compatible with digital input circuits, which permits multiplexing of digital output points with the user's contacts for digital input sensing applications. Either positive or negative voltage sources can be used through proper selection of the input terminals when the digital output medium-power group or digital output contact group is used. This circuit compatibility also permits direct connection of digital input points to digital output points for testing of the input/output circuits.

I/O Commands

Data transfers between the digital output function and the processor are accomplished by immediate read and immediate write commands. The setting of the modifier field bits in the command further defines the operation to be performed.

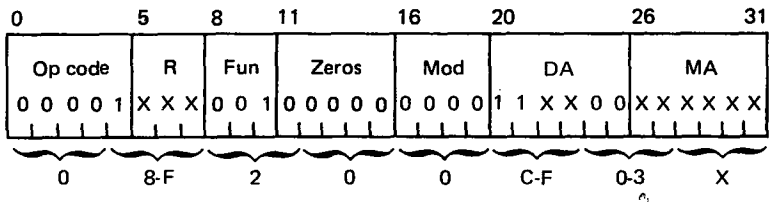
All read and write I/O commands addressed to the digital output control must use one of the following device addresses in order to address the corresponding digital output group separately:

Device Address	*Digital Output Group*
110000	0
110100	1
111000	2
111100	3

The module address in the I/O command varies depending upon the physical location of the multifunction module in the 5026 Enclosure.

The normal digital output programming routine is shown in Figure F-5 of "Appendix F."

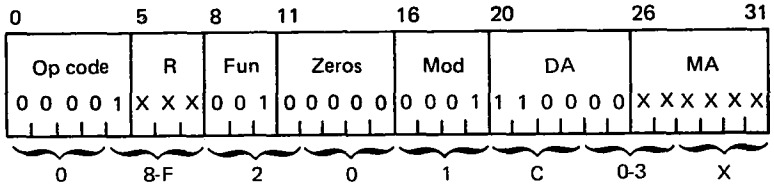
Write Digital Output Group Register



This command stores 16 bits of output data from the index register (R), or the accumulator if R=000, into the digital output group register identified by the device address field.

The status of the 16-bit digital output register is reproduced at the corresponding terminals for its associated 16-point group. A 1-bit in the register produces a closed or on condition at its corresponding output point; a 0-bit in the register produces an open or off condition at its corresponding output point.

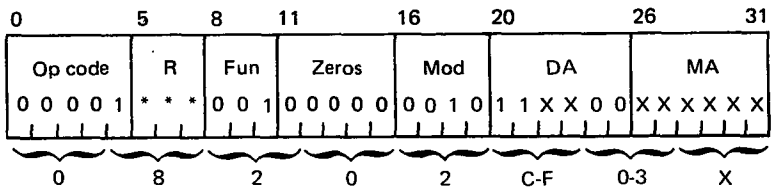
Write Digital Output Holding Register



This command stores 16 bits of output data from the index register (R), or the accumulator if R=000, into the holding register.

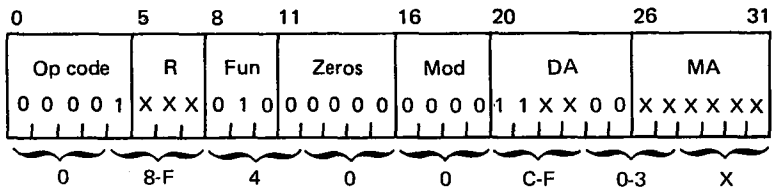
The holding register is a buffer that allows the data to be verified by the program prior to transferring it to one of four output registers (and to the corresponding terminals).

Set Digital Output Group Register



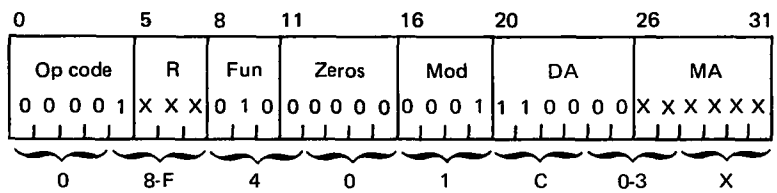
This command transfers the 16 bits of output data in the holding register to the digital output group register identified by the device address field. The R field is ignored.

Read Digital Output Group Register



This command stores the 16 bits of output data from a group register (identified by the device address field) into the index register (R), or the accumulator if R=000.

Read Digital Output Holding Register



This command stores the 16 bits of output data from the holding register into the index register (R), or the accumulator if R=000.

ANALOG OUTPUT CONTROL

The Analog Output Control feature of the multifunction module has either one or two isolated analog output voltage points. Each point provides a voltage output in the range of either 0V to +10.23V or 0V to −10.23V, depending on the wiring polarity of the output point at the user's terminals.

The analog output points can be operated by either of two methods: (1) data can be sent directly to a selected analog output register which, in turn, operates an output point; or (2) data can be sent to the holding register and then read back and checked for accuracy before transferring it to one of two analog output registers.

The analog output function can be used as input to the analog input function, which permits multiplexing of analog output points with the user's equipment for analog input sensing applications.

Caution must be used when connecting analog input circuits directly to analog output circuits because the analog output function can produce a voltage greater than the maximum allowable input voltage to the analog input function. For this reason, a 2:1 divider is recommended to prevent overload and possible damage to the analog input circuits. (See Appendix E for conversion of analog voltage to its binary equivalent.)

I/O Commands

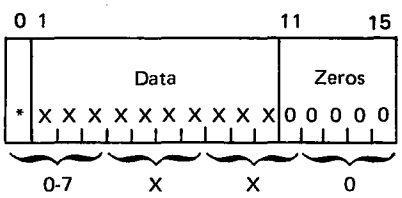
Data transfers between the analog output function and the processor module are accomplished by immediate read and immediate write commands. The setting of the modifier field bits in the command further defines the operation to be performed.

All read and write I/O commands addressed to the analog output control must have one of the following device addresses in order to address the corresponding analog output point separately:

Device Address	*Analog Output Point*
101000	0
101100	1

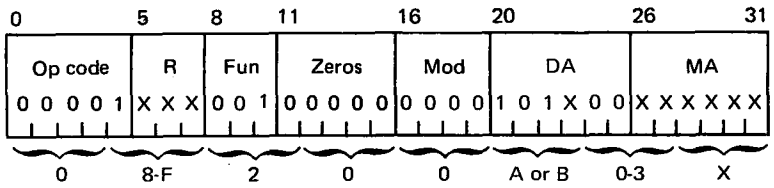
The module address in the I/O command varies depending upon the physical location of the multifunction module in the 5026 Enclosure.

Analog output data has a resolution of only 10 bits and has the following format for all commands that use the R field of the I/O command:



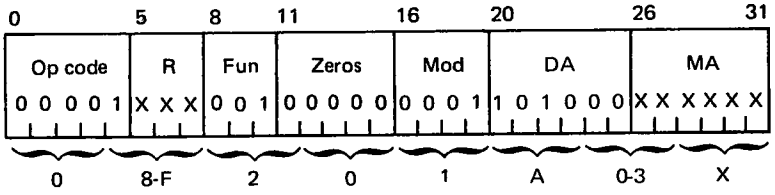
The normal analog output programming routine is shown in Figure F-2 of "Appendix F."

Write Analog Output Register



This command stores the output data from the index register (R), or the accumulator if R=000, into the analog output register identified by the device address field. The contents of the 10-bit analog output register are then converted to the corresponding voltage signal, which is reflected at the addressed analog output point terminals.

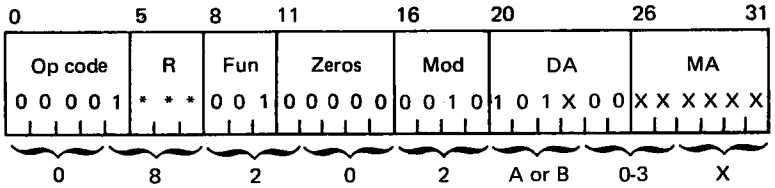
Write Analog Output Holding Register



This command stores the output data from the index register (R), or the accumulator if R=000, into the holding register.

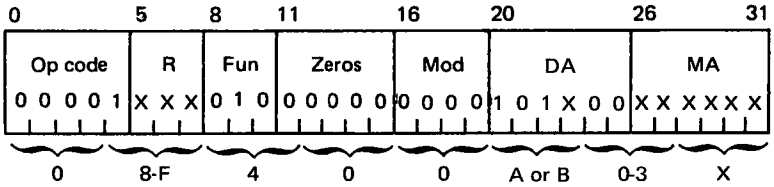
The holding register is a buffer that allows the data to be verified by the program prior to transferring it to one of two analog output registers (and to the corresponding terminals).

Set Analog Output Register



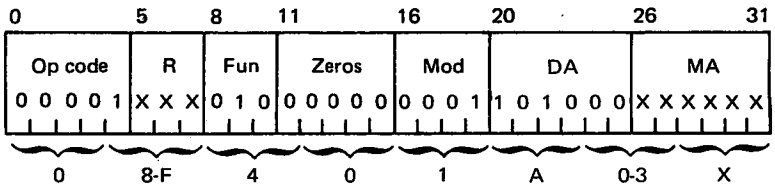
This command transfers the output data in the holding register to the analog output register identified by the device address field. The contents of the 10-bit analog output register are then converted to the corresponding voltage signal, which is reflected at the addressed analog output point terminals. The R field is ignored.

Read Analog Output Register



This command stores the output data from an analog output register (identified by the device address field) into the index register (R), or the accumulator if R=000. Bits 11 to 15 of the R register are set to 0's.

Read Analog Output Holding Register



This command stores the output data from the holding register into the index register (R), or the accumulator if R=000. Bits 11 to 15 of the R register are set to 0's.

2790 CONTROL

The 2790 Control feature of the multifunction module attaches a single loop of the IBM 2790 Data Communication System to the System/7 as shown in Figure 13-1. This feature provides the user with a flexible plant communication and data collection system.

The 2792 Remote Communications Controller attaches a remote loop to the 2790 single loop as shown in Figure 13-1. This capability extends the System/7's locally controlled 2790 Data Communications System to remote locations via a full duplex (with four-wire termination) leased common-carrier line. The transmission rate is 2400 bits per second.

The 2790 system provides for rapid transfer of digital information from data entry units at various plant locations to area stations at key locations and on to a system controller. The System/7 is the system controller for a 2790 system attached to the 2790 Control in the 5012 Multifunction Module. Data enters the 2790 system from a card reader, badge reader, pulse counter, or key entry unit. Output data to the 2790 system is sent to 2791 indicator lights, to the 2798 display and guidance panel, or to a printer attached to either a 2791 or 2793 Area Station.

The 2790 Data Communication System is connected by a two-wire, high-speed loop capable of operating at about 500,000 bits per second (about 900 characters per second) partitioned over the active terminals on the loop. This loop is a serial connection that starts and ends at the 2790 Control feature in the System/7.

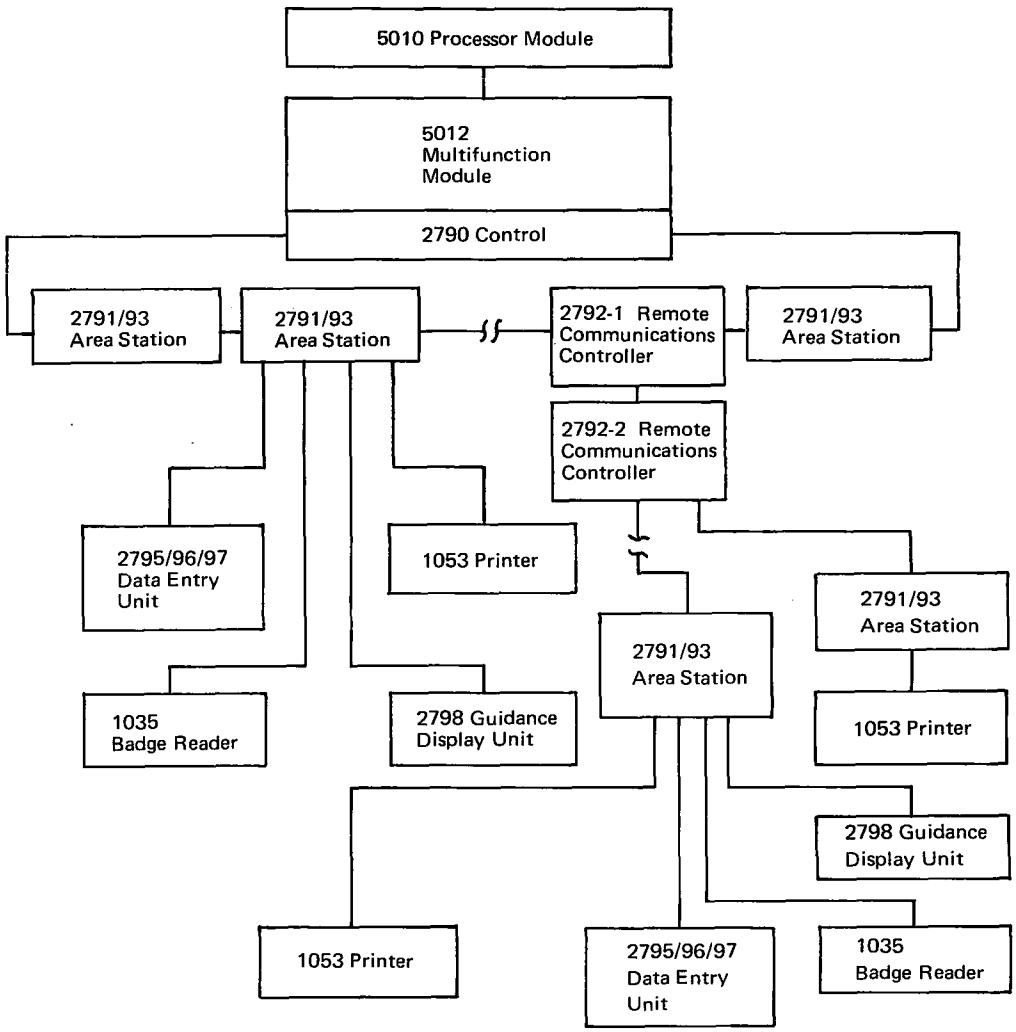


Figure 13-1. 2790 Data Communication System attached to System/7

Multiple area stations (2791 and/or 2793) attach to the transmission loop by inserting a one-byte shift register in series with the line. This shift register (Figure 13-2) enables the area station to examine and modify a full byte of information before passing it on to the next area station in the loop. Multiple 2795/2796/2797 Data Entry Units, 1035 Badge Readers, 2798 Guidance Display Units, and 1053 Printers can be connected to each area station. The maximum distance between any two area stations on the loop, and between the first/last area station and the 2790 Control, is governed by the type of wire used. See Appendix B in the manual *IBM 2790 Data Communication System: Installation Manual–Physical Planning,* Order No. GA27-3017. Remote locations may be serviced via a remote loop attached through a 2792 pair.

Programming and storage requirements to support the 2790 system limit the actual configuration of area stations and associated input and output units that may be attached to the loop. These limits are specified in manuals that describe programming support for the particular system using the programs. Programming support is provided for up to four 2790 Control Features in a System/7 configuration.

For more detailed information on the capabilities, components, features, and applications of the 2790 Data Communication System, refer to the manuals *IBM 2790 Data Communication System Component Description,* Order No. GA27-3015, and *IBM 2790 System Summary,* Order No. GA27-3016.

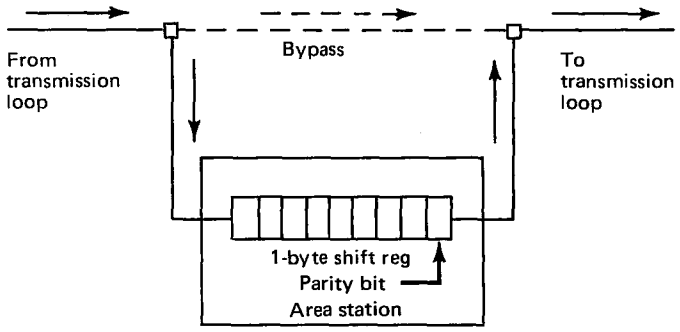


Figure 13-2. One-byte shift register

Data Transmission

Information is transmitted over the loop by data bits, represented as bipolar pulses at approximately 500 kHz, as long as the 2790 Control in the 5012 Multifunction Module has power. All communication between the System/7 and the area stations is accomplished by a sequence of five information bytes, called a frame, shown in Figure 13-3. Byte 0 (the start byte) is generated by the 2790 Control; the remaining four bytes (area station address, device address, control, and data) are generated by the program in the processor module. Figure 13-3 shows that 25 synchronization bytes separate each five-byte active portion in a frame cycle, for a total of 30 bytes. These 30 bytes make up what is called an active frame. All synchronization bytes are generated by the 2790 Control.

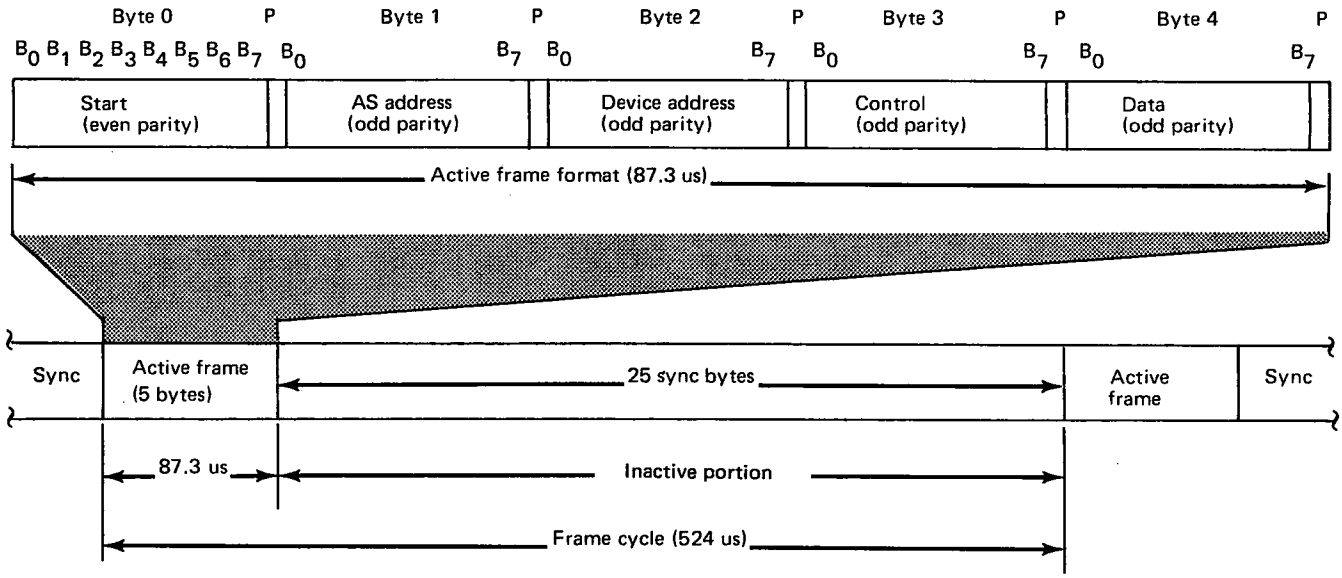


Figure 13-3. Data transmission format

Frames are transmitted by the 2790 Control at intervals of 524 microseconds. Active frames can be separated by one or more inactive frames, which consist of a start byte and 29 synchronization bytes. If the controlling program does not specify a new active frame, an inactive frame is sent out by the 2790 Control. This separation is used primarily to allow time to process input frames and to prepare output frames. The frame fields are summarized in Figure 13-4.

When multiple 2790 Controls (up to 4) are installed in a System/7 configuration, the output frames transmitted by each 2790 Control are offset by 3 bytes from the output frames transmitted by any other 2790 Control. In other words, the first 2790 Control starts transmitting output frames. When byte 3 of its first output frame is sent, then the second 2790 Control starts transmitting. When byte 3 of its first output frame is sent, then any third 2790 Control starts transmitting. When byte 3 of its first output frame is sent, then any fourth 2790 Control starts transmitting. Once begun, the transmission of output frames from all 2790 Controls is continuous.

Note: When active frames are transmitted too rapidly to the same device, critical timing conditions may occur. These conditions are described later in this chapter under "Critical Timing Conditions."

Field	Byte	Hex	Bit pattern 0 1 2 3 4 5 6 7 P
Start	0	39	0 0 1 1 1 0 0 1 0
AS address	1		
Discrete AS		80-FF	1 A A A A A A A -
All AS		09	0 0 0 0 1 0 0 1 1
Any AS		11	0 0 0 1 0 0 0 1 1
Software idle		18	0 0 0 1 1 0 0 0 1
Device address	2		Select Adapter
Diagnostics		00	0 0 0 0 0 0 0 0 -
Pulse counter		01-3F	0 0 A A A A A A
Data entry unit		C0-DF	1 1 0 A A A A A -
Local output			
1053 Printer		40	0 1 0 0 0 0 0 0 0
Area station display		80	1 0 0 0 0 0 0 0 0
Local input			
Badge reader		80	1 0 0 0 0 0 0 0 0
Card reader		84	1 0 0 0 0 1 0 0 1
Keyboard		88	1 0 0 0 1 0 0 0 1
OEM		8C	1 0 0 0 1 1 0 0 0
1035 Badge Reader			1 0 0 0 0 0 A A -
Remote X		81	1 0 0 0 0 0 0 1 1
Remote Y		82	1 0 0 0 0 0 1 0 1
Remote Z		83	1 0 0 0 0 0 1 1 0
Guidance Display Unit		90-9F	1 0 0 1 A A A A -
Control (Refer to Figure 13-5 for all command/response bit patterns)	3		AS Response digit / Command digit U V W W Y Y Z Z Op code Modifier AS defined Status bit Data mode Restore Ack
Data, status, or guidance	4		D D D D D D D D -
Sync (hardware idle)	5-29	47	0 1 0 0 0 1 1 1 1

"A" denotes an address bit

"AS" denotes area station

Figure 13-4. Frame field code structure

The functions of the five bytes in an active frame are described in the following paragraphs.

Start

The start byte, which contains hexadecimal 39, signals the start of a frame cycle. The end of an active frame is indicated by the selected area station reaching a byte count of five. The start byte, generated by the 2790 Control, is the only byte that requires even parity.

Area Station Address

Coded information in this byte can be used to select a unique area station, any area station, or all area stations.

There are 128 unique area station addresses, which must be indicated by codes within the range of hexadecimal 80 to hexadecimal FF. All address codes outside this range are invalid. Because of the characteristics of the system, it is recommended that area station addresses be assigned sequentially starting with hexadecimal 80. This does not, however, restrict the physical placement of area stations on the loop.

An any-area-station address is designated by a code of hexadecimal 11. Whenever an area station requires service from the program, it searches for a frame having this code. Upon finding one, the area station captures the frame by inserting its own address in place of the any-area-station address in this byte.

An all-area-stations address is designated by a code of hexadecimal 09. Every area station executes the instruction specified in the control byte, if it can, without modifying the area station address byte.

A software idle is designated by a code of hexadecimal 18. This code is used by the 2790 Control programming to reserve a specific frame. No action is taken by the area station.

Device Address

A frame addressed to an area station is also addressed to a particular device (such as a 2795 Data Entry Unit) on a specific adapter connected to that area station. Bits 0 and 1 of the device address byte identify the particular adapter in the area station, or determine that this byte is a part of diagnostic controls. Bits 2 to 7 identify the device attached to the addressed adapter. The diagnostic controls are normally addressed by hexadecimal 00. However, a diagnostic command is executed regardless of the contents of this byte. If an area station adapter is selected, or an area station captures an any-address frame, the area station inserts its own address and the address of a particular device into the frame.

Control

The control byte is divided into two hexadecimal digits. The low-order digit (bits 4 to 7) contains the command issued by the 2790 Control. The command digit is further subdivided into an operation code (bits 6 and 7) and a modifier (bits 4 and 5).

The high-order digit (bits 0 to 3) contains the response from an area station. An addressed area station always responds to a command that is valid for that particular area station. Bit 0 of the response digit is the restore acknowledge bit and is nearly always set to 0. Bit 1 is the data mode bit. It is set on when the area station decodes one of the diagnostic commands, or when the addressed adapter is in the data mode and decodes a valid operation code. Bits 2 and 3 are status bits, and are defined by the area station to reflect the status of an operation in the area station adapter.

The control byte codes permit two-way communication between the area stations and the system controller for every step during the various operations.

Data

The data byte contains either data or status information, depending upon the contents of the control byte.

A data transfer caused by a read command always involves receiving the data byte twice from the area station, once in each of two frames. Thus, input data can be checked for accuracy without special checking logic in the area station.

Data transferred as a result of a write command is also checked for accuracy by transmitting it to an area station and then receiving the data back from the area station. If the data received is the same as the data transmitted, the area station is then commanded to print the data or turn on a specified indicator light.

Commands and Responses

The control byte of each transmission frame contains coding and responses to all commands. All commands, which are generated by the System/7 program, are sent from the 2790 Control to an area station, whereas all responses are returned from an area station to the 2790 Control. Thus, a control byte transmitted from the 2790 Control should contain a response when that command is received in the 2790 Control.

Each command and response is represented by two hexadecimal digits corresponding to the coding within the eight-bit control byte. The first hexadecimal digit (four high-order bits of the control byte) is called the area station response digit. The second hexadecimal digit (four low-order bits of the control byte) is called the command digit. This command digit is further divided into a two-bit modifier and a two-bit op code as shown in Figure 13-5.

Bit 0 of the response field is turned on to indicate that a frame is being sent by the 2792 to the remote loop and a response is pending.

Command frames are easily distinguished from response frames because the response digit is zero for all commands, but nonzero for all responses as shown in Figure 13-5. There are several commands in each of the three major command categories: read, write, and control. In addition, there is at least one valid response for each of the commands.

Note: When power is initially applied to the area stations, they are in the bypass mode of operation. (This mode is explained later in this chapter.) The controlling program must issue control commands to restore discrete area stations, or a single control command addressed so as to restore all area stations. The restore operation inserts the area station(s) into the transmission loop.

Code name	Hex	Bit pattern	
		Response 0123	Command 4567
Read commands (from System/7)			
Read	06	0000	0110
Read null	02	0000	0010
Read data	0A	0000	1010
Read end	0E	0000	1110
Read responses (from area station)			
Read request	12	0001	0010
Read cmd acknowledge	46	0100	0110
Read data request	62	0110	0010
Read data acknowledge	6A	0110	1010
Read end request	72	0111	0010
Read end acknowledge	3E	0011	1110
Read null acknowledge	42	0100	0010
Write commands (from System/7)			
Write	05	0000	0101
Write null	01	0000	0001
Write data	09	0000	1001
Write end	0D	0000	1101
Write responses (from area station)			
Write data request	51	0101	0001
Write data acknowledge	69	0110	1001
Write end request	71	0111	0001
Write end acknowledge	3D	0011	1101
Write null acknowledge	41	0100	0001
Write cmd acknowledge	45	0100	0101
Control commands (from System/7)			
Bypass	0B	0000	1011
Restore	03	0000	0011
Send sync	04	0000	0100
Begin diagnostic	08	0000	1000
End diagnostic	0C	0000	1100
2792 Reset	0F	0000	1111
Control responses (from area station)			
Bypass acknowledge	6B	0110	1011
Restore acknowledge	63	0110	0011
(response to send sync cmd)	47	0100	0111
Begin diagnostic acknowledge	68	0110	1000
End diagnostic acknowledge	4C	0100	1100

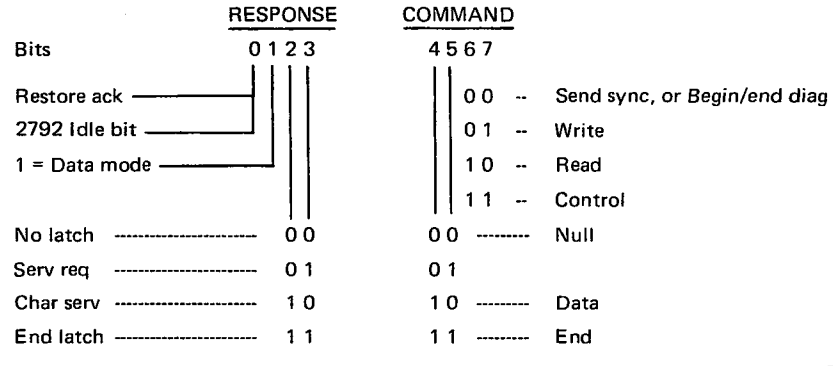


Figure 13-5. 2790 loop commands and responses

Read Commands

All read commands are generated by the program in the System/7 processor and sent to the area station by the 2790 Control.

Read: This code (hexadecimal 06) is sent after an area station has captured an any-address frame and the area station has returned a read request response. The code puts the area station adapter in data mode. The only valid response is read command acknowledge.

Read Null: This code (hexadecimal 02) specifies that the frame is available for any type of read request. This code should be generated by the program when no other command is to be transmitted. Four valid responses are: read null acknowledge, read request, read data request, and read end request.

Read Data: This code (hexadecimal 0A) is sent when a read data request response is received. The code causes the area station adapter to send the data byte again, so that a validity check can be performed, and also releases the area station adapter to prepare for the next character. The only valid response is read data acknowledge.

Read End: This code (hexadecimal 0E) is sent either as a result of receiving a read end request from the area station adapter or as a result of detecting an error that requires the operation to be terminated. This command removes the area station adapter from data mode, thus causing the operation in progress to be terminated in the area station adapter. The area station refers to the data byte to determine which indicator light to turn on. The only valid response to this command is read end acknowledge. When the command is sent to an area station local I/O adapter, the data byte contains an operator guidance character. (Refer to the manual *IBM 2790 Data Communication System Component Description,* Order No. GA27-3015, for a description of the operator guidance functions.)

Read Responses

All read responses are generated by the area station and sent back to the 2790 Control.

Read Request: This code (hexadecimal 12) is a response to a read null command when the area station captures an any-address frame to initiate a read operation. The code indicates that the area station wants to transfer information to the 2790 Control.

Read Command Acknowledge: This code (hexadecimal 46) is a response to a read command. The response indicates that the area station has received the read command and will start to read its data source (for example, a badge or a data entry unit).

Read Data Request: This code (hexadecimal 62) is a response to a read null command if the area station adapter has a data byte to send to the 2790 Control.

Read Data Acknowledge: This code (hexadecimal 6A) is a response to a read data command. The response indicates that the area station has executed the command. The area station response digit is the same as in read data request response. The controlling program should check the data byte received for validity against the one received earlier with the read data request.

Read End Request: This code (hexadecimal 72) is a response to a read null command if the area station adapter has determined either that the data transfer has come to a normal completion or that an error has been detected. The status of the operation is transmitted in the data byte. An all-zero status designates a normal end.

Read End Acknowledge: This code (hexadecimal 3E) is a response to a read end command. The response indicates that the area station adapter will terminate the operation according to the status in the data byte. The transaction is completed when the code is received by the 2790 Control. The program should check the data byte received (ending status) for validity against the one transmitted earlier with the read end command.

Read Null Acknowledge: This code (hexadecimal 42) is a response to a read null command if no action is presently required by the area station adapter and the area station is in data mode.

Write Commands

All write commands are generated by the program in the System/7 processor and sent to the area stations by the 2790 Control.

Write: This code (hexadecimal 05) initiates a write operation and puts the area station adapter in data mode. The only valid response is write command acknowledge. If the area station adapter cannot execute this command, it will make an end request when it receives the next command, a write null command. If a write command is sent to an area station local I/O adapter and the adapter is busy, the area station will return a data byte containing hexadecimal 80. The command should be sent repeatedly until a write command acknowledge response is received.

Write Null: This code (hexadecimal 01) specifies that the frame is available for any type of write request from the device that is addressed. There are three valid responses: write null acknowledge, write data request, and write end request.

Write Data: This code (hexadecimal 09) is sent to an area station when a write data request is received. This command is accompanied by a data byte, and the only valid response is write data acknowledge.

Write End: This code (hexadecimal 0D) is normally sent if a write data request response is received and there is no data left to be transmitted. The command can also be sent as a result of a program detected error, or as a result of a write end request response by the area station adapter after it detects an abnormal condition. The data byte with the write end command is used as a status field, with all 0's representing normal end. The only valid response is a write end acknowledge.

Write Responses

All write responses are generated by the area station and sent back to the 2790 Control.

Write Data Request: This code (hexadecimal 51) is a response to a write null command if the area station adapter is ready to receive a data byte.

Write Data Acknowledge: This code (hexadecimal 69) is a response to a write data command, and indicates that the area station has received the data byte. The program should check the data byte received for validity against the one transmitted earlier with the write data command.

Write End Request: This code (hexadecimal 71) is a response to a write null command if the area station adapter has detected an error or an unusual condition. The status is also sent in the data byte.

Write End Acknowledge: This code (hexadecimal 3D) is a response to a write end command. This response indicates that the area station adapter will terminate the operation according to the status in the data byte as modified by the area station adapter. An all-zero status represents normal end. The received status byte should be checked by the program for validity against the transmitted status byte.

Write Null Acknowledge: This code (hexadecimal 41) is a response to a write null command if no action is presently required by the area station adapter and the adapter is in the data mode.

Write Command Acknowledge: This code (hexadecimal 45) is a response to a write command. The response indicates that the area station adapter has received the command and that the adapter has inserted its status in the data byte. An all-zero status indicates that the operation will be executed. A nonzero status indicates busy, which can be returned only by the area station local I/O adapter.

Control Commands

All control commands are generated by the program in the System/7 processor and sent to the area stations by the 2790 Control.

Bypass: This code (hexadecimal 0B) can be used by the program as part of its diagnostic procedures. The command causes the area station to remove its shift register from the loop after the control byte has been received. This bypass operation eliminates the data byte from this frame, since there is a one-byte delay in the shift register. Since the area station removes the data byte from the frame, a short count results in the 2790 Control byte counter. However, the start byte of the next frame resets the byte counter.

While an area station is in a bypassed state, it can monitor the line, but it cannot insert information. The only valid response is bypass acknowledge.

Restore: This code (hexadecimal 03) is used by the program as part of its diagnostic procedures. The command causes the bypassed area station to insert its shift register in series with the loop after the control byte has been received. This insertion results in a repeat of the control byte (the restore command), which adds one byte to the active portion of the frame. A skip character (all 0's) is also substituted for the old data byte. The active frame returning to the 2790 Control contains six bytes instead of five. The fourth and fifth bytes contain the restore command and the sixth byte contains the skip character. The skip character causes the 2790 Control byte counter to skip a count.

If a restore command is sent to an area station that is not bypassed, the area station will return a restore acknowledge response, which enables the program to determine whether the command has been executed. The only valid response is the restore command itself immediately followed by a repetition of the restore command.

Send Sync: This code (hexadecimal 04) may be used by the program after an area station has been placed in diagnostic mode. When an area station in diagnostic mode receives a send sync command, the area station sends a continuous stream of sync characters (each containing hexadecimal 47) until it receives from the 2790 Control at least two characters in succession containing hexadecimal FF. This can be achieved by sending a frame with hexadecimal FF in all byte positions. The only response to this command is the fact that the area station generates a stream of sync characters.

The send sync command provides the ability to isolate a link (connecting two successive area stations) that is failing intermittently and causing the system's maximum error rate count to be exceeded.

The command allows predetermined data (sync characters) to be constantly transmitted by a selected area station without that area station transmitting what it is receiving.

2792 Reset: This code (hexadecimal 0F) is used to reset circuitry in the 2792 for initial startup and error reset conditions. It is recognizable only when used with an "all area station" address (hexadecimal 09) and is ignored by 2791/93 area stations.

Begin Diagnostic: This code (hexadecimal 08) can be used by the program as part of the area station diagnostic procedure. The addressed area station enters diagnostic mode when it receives this command. In diagnostic mode, the area station responds only to commands addressed to it. The only valid response is begin diagnostic acknowledge.

End Diagnostic: This code (hexadecimal 0C) is used by the program to end the area station diagnostic procedure. The area station is removed from diagnostic mode and returns to normal operation when this command is received. The only valid response is end diagnostic acknowledge.

Control Responses

All control responses are generated by the area station and sent back to the 2790 Control.

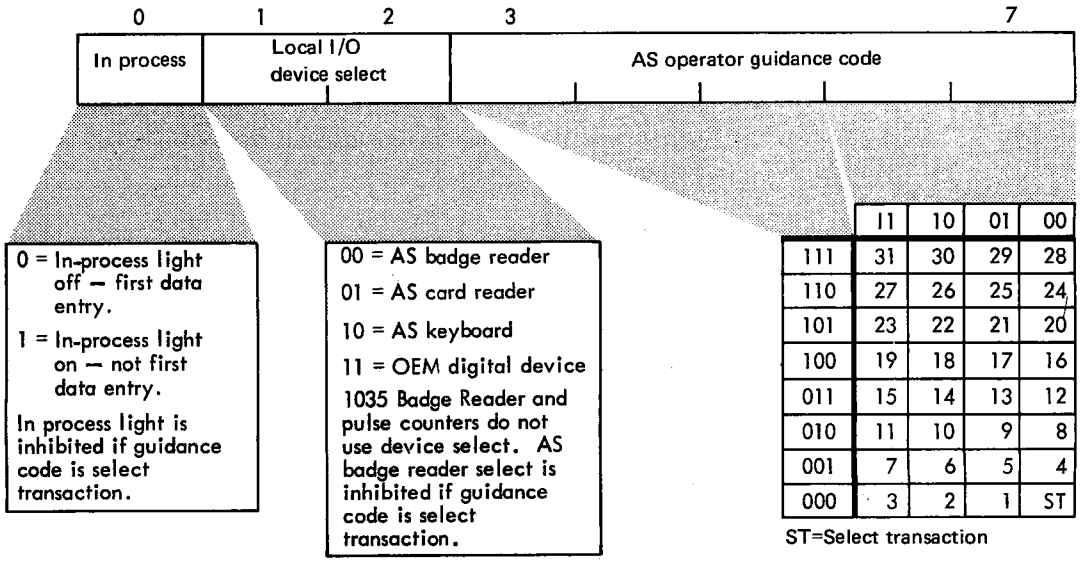
Bypass Acknowledge: This code (hexadecimal 6B) is a response to the bypass command and indicates that the area station has executed the command.

Restore Acknowledge: This code (hexadecimal 63) is a response to the restore command if the area station is already on line (not bypassed) when the command is received. This response is also inserted prior to the data byte if the area station is off line (bypassed) when the restore command is received.

Begin Diagnostic Acknowledge: This code (hexadecimal 68) is a response to the begin diagnostic command, and indicates that the area station has entered diagnostic mode.

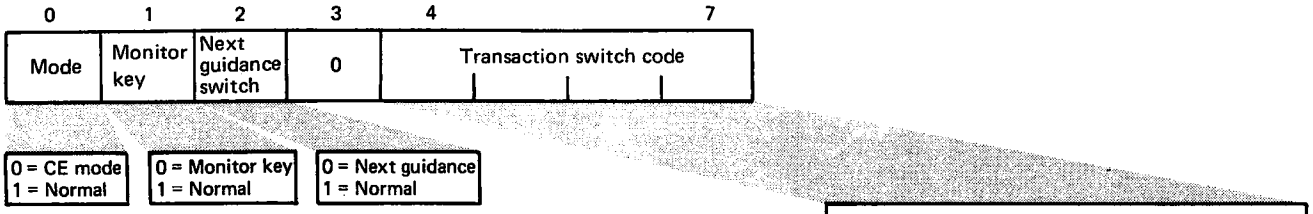
End Diagnostic Acknowledge: This code (hexadecimal 4C) is a response to the end diagnostic command, and indicates that the area station has returned to the normal mode of operation.

Area Station Local I/O Guidance Character



The guidance character is the second character transmitted in any message from the AS local I/O devices (except 1035 Badge Reader and pulse counters). In this case, the the guidance character reflects the "old guidance"--that is, the previous guidance character sent with a read end command to a local I/O device. A guidance character sent with a read end command to a local I/O device is considered to be "new guidance." It becomes "old guidance" when a new message is initiated.

Area Station Local I/O Transaction Code Byte



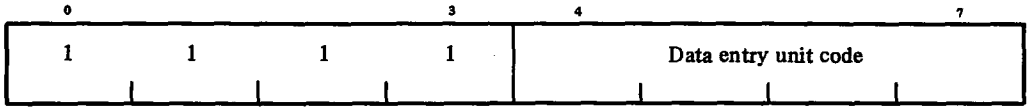
This transaction code is the first data character transmitted in any message from the AS local I/O devices.

These devices include:

- AS badge reader
- AS card reader
- AS keyboard
- 1035 Badge Readers
- AS transaction and next guidance switches
- OEM digital device

0000 = No transaction
0001 = Switch 1
0010 = Switch 2
0011 = Switch 3
0100 = Switch 4
0101 = Switch 5
0110 = Switch 6
0111 = Switch 7
1000 = Switch 8
1001 = Switch 9
1010 = 1035 Badge Readers
1111 = CE transaction

Area Station Local I/O Guidance Character



Status character codes

Status character (hex)	*Associated with end request from area station means:*	*Associated with end command results in the following action at the area station:*
	Local I/O of area station	
00	Good end	Select transaction indicator
20	Diagnostic mode	No action
40	Release in data mode	Light repeat/clear
60	Release in diagnostic and data mode	Light repeat/clear
80	Invalid character	Light repeat/clear
A0	Invalid character and diagnostic mode	Light repeat/clear
C0	Overrun—card reader	Light repeat/clear
E0	Overrun in diagnostic mode	Light repeat/clear
All other	Accepted as guidance characters by local I/O device	

Status character (hex)	*Associated with end request from area station means:*	*Associated with end command results in the following action at the area station:*
	Data entry unit	
00	Good end	Release work station
20	Diagnostic mode	No action
40	Overrun	Error indication
60	Overrun in diagnostic mode	No action
80	EOB time-out	Error indication
A0	EOB time-out and diagnostic mode	No action
	Output adapter	
00	Good end	No action
20	Diagnostic mode	No action
80	Out of paper	No action
A0	Out of paper and diagnostic mode	No action

Critical Timing Conditions

Input/output adapters contained in 2791/2793 Area Stations are designed to operate at a data rate of 212 frames per second, spaced at equal intervals of 4.7 milliseconds. However, the time required for mechanical operations to take place within a 2790 system device may reduce the data rate from the maximum of 212 frames per second.

Figure 13-6 shows the 2790 devices that have critical timing conditions. Most of these critical timings apply only to a specific sequence of commands when they are addressed to the particular device. For example, a read data command frame addressed to a 1035 Badge Reader must be followed by two frames that are not addressed to the same 1035 Badge Reader before a read null command frame can be addressed to this badge reader. (A read null command frame can, however, be addressed to another device or another 1035 Badge Reader.)

Critical timing situations can be avoided by observing the designed data rates. These rates can be achieved by using every 45th frame to communicate with a 1053 Adapter, and every 9th frame to communicate any command sequence to other input/output adapters.

2791/2793 Adapter	Critical timing
Input adapter 1035 Badge Reader	Minimum of 2 frames* between a read data command frame and a read null frame.
Card reader	① Minimum of 2 frames* between a read data request frame and read data command frame. ② Maximum of 14 frames* between two consecutive read null frames (to avoid character overrun condition in 2790).
Data entry unit adapter	Maximum of 16 frames* between two consecutive read null frames (to avoid character overrun condition in 2790).
1053 Adapter	① Normal rate of 42 frames per second at equal intervals of 23.5 milliseconds. (Approximately 45 frames* between frames addressed to this adapter). ② Minimum of 133 frames* between two consecutive write null command frames that follow a write data command frame.

* Frames included in this count cannot address the adapter or device to which this timing applies.

Figure 13-6. 2790 critical timings

2792 Error Status

Status Characters (hex)	Meaning
00	Good status
01	Open loop
02	Data check
04	Control check
08	Parity error
10	Communications data error
20	Communications channel timeout
40	All area station command error

2790 Control Operation

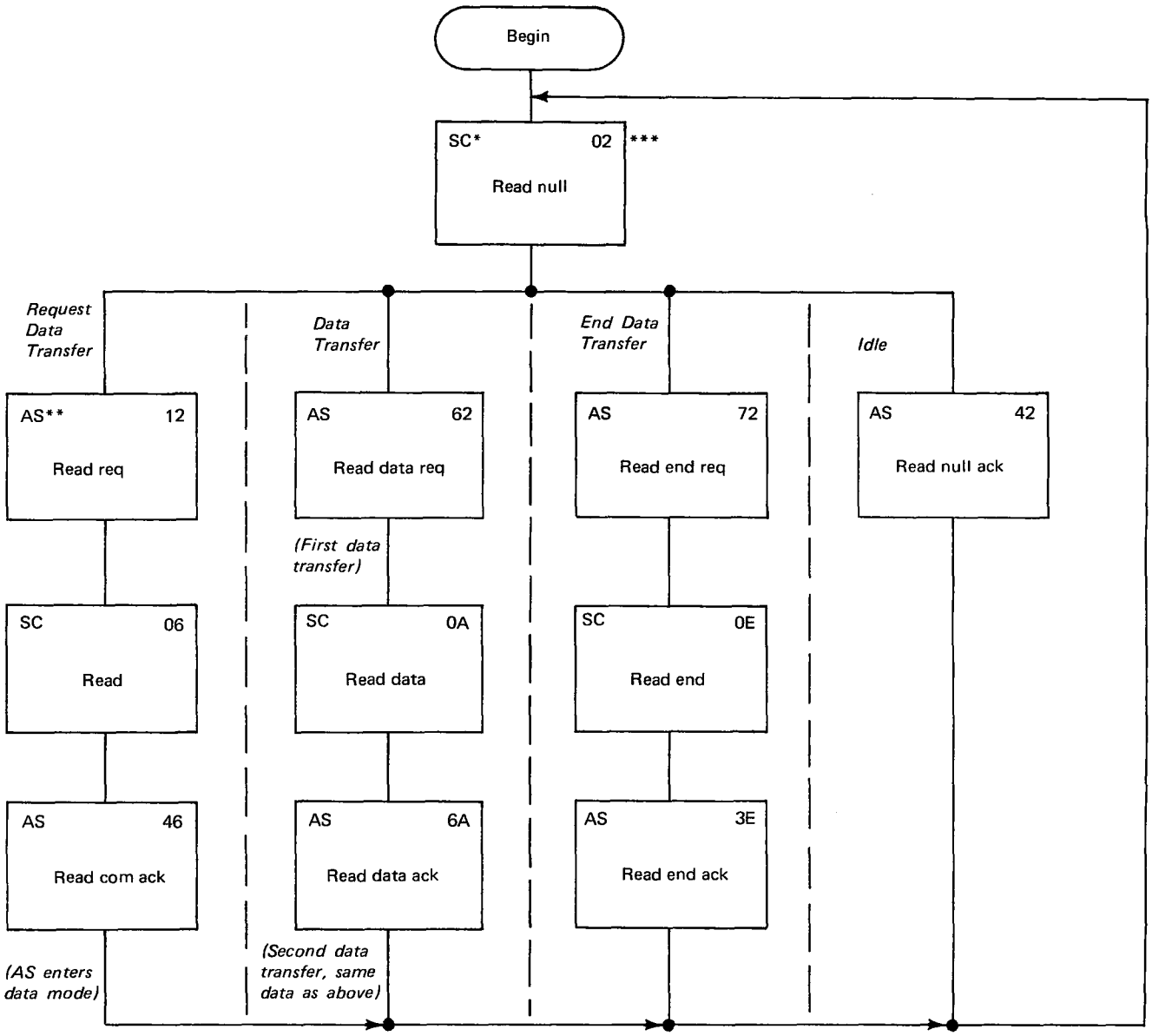
The 2790 Control uses two 4-byte buffers—an output buffer and an input buffer. The output buffer must be loaded with the area station address, device address, control, and data bytes to be transmitted on the 2790 loop. When the loop is not actively transmitting data (the four output buffer registers have not been filled by the program), a start byte followed by 29 sync bytes will be transmitted on the loop every 524 microseconds. When 30 bytes have been transmitted and the output buffer register has been filled, the next outgoing frame contains the start byte followed by the contents of the four-byte output buffer register. Twenty-five sync bytes follow this active frame, and then another start byte.

Frame cycles transmitted from the output buffer are received back in the 2790 Control in the four-byte input buffer. When the area station address, device address, control, and data bytes are received in the input buffer, the 2790 Control presents an interruption request to the System/7 processor. Errors are detected in the 2790 Control at the input buffer register.

Command Sequences

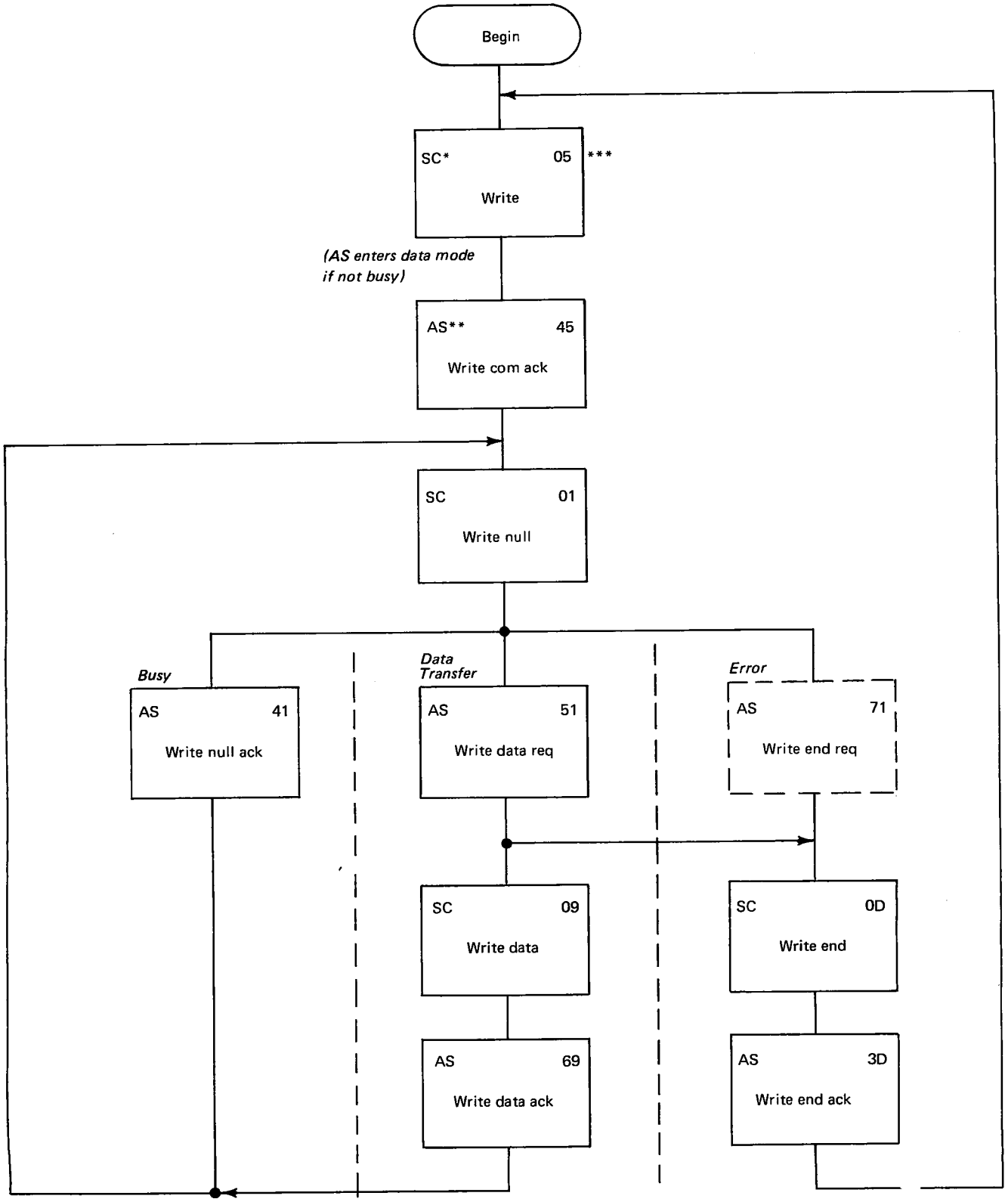
The 2790 Control must transmit several active frame cycles to perform a single operation. Figures 13-7 and 13-8 show read and write command sequences, respectively, and the proper area station responses to the commands. Critical timing conditions may exist for certain 2790 devices as shown in Figure 13-6.

For example, when no area stations require servicing, a frame containing an any-address code in the area station address byte and a read null command in the control byte is transmitted on the loop (see Figure 13-7).



*SC = 2790 control program command
**AS = area station response
*** = hexadecimal code

Figure 13-7. 2790 read sequences



*SC = 2790 control program command
**AS = area station response
*** = hexadecimal code

Figure 13-8. 2790 write sequences

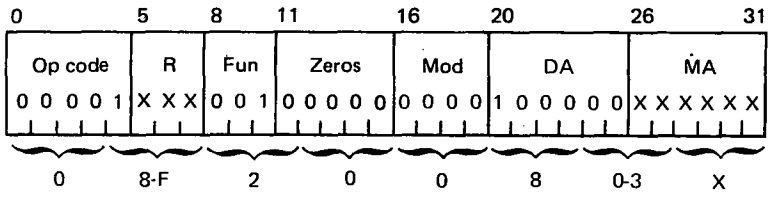
An area station requiring service looks for the any-address code and captures the frame by replacing the code with its discrete area station address. This not only identifies the area station requesting service, but also prevents any other area station from capturing this frame. The area station then inserts the address of the device that will be serviced into the device address field and inserts a read request response code in the control byte. The program then generates an outgoing frame containing the area station address, device address, a read command, and a 0 data byte. This transmission places the area station in data mode, which permits the requesting device to transmit data. The area station responds to the read command, and the program sends out a read null command. The area station responds to the read null command with a read data request when it has a data byte ready to be sent to the 2790 Control. This data byte accompanies the read data request response. The program then sends out a read data command, to which the area station responds with a read data acknowledge and, again, the data byte. This double transmission of the data byte allows for validation of every data byte transmitted by the area station.

Under normal operation the area station can initiate the read-end sequence when it has completed a data or transaction code entry. The sequence is started when the area station responds to a read null command addressed to it with a read end request and normal status in the data byte. The controlling program then sends out a read end command with either a guidance character or status information in the data field, depending on whether the addressed device is a data entry unit or a device on the area station local I/O adapter. Guidance characters apply only to the local I/O adapter device.

I/O Commands

The prepare I/O command is used as described in Chapter 4 under "Input/Output Instruction." Data transfers between the 2790 Control and the processor module are accomplished by immediate read and immediate write commands. Modifier field bits in the immediate read and write commands further define the operation to be performed. All I/O commands addressed to the 2790 Control function must have a device address of 100000. The module address varies, depending upon the physical location of the multifunction module in the 5026 Enclosure. The 2790 Control has two 4-byte buffer registers, one for input frames and one for output frames. Two immediate write commands are used to fill the output buffer register with a frame for transmission, while two immediate read commands are used to read the frame in the input buffer register. The start byte (the first byte of a frame) is generated by the 2790 Control, so this byte is not loaded into the output buffer or read from the input buffer.

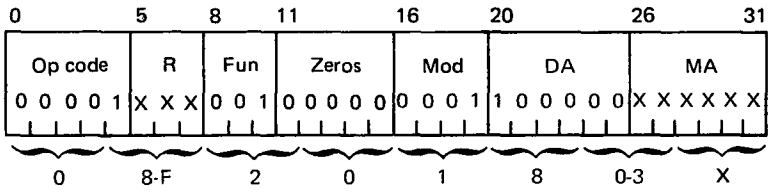
Write Address



This command stores the area station address and device address bytes into the first two bytes of the output buffer register in the 2790 Control. These two address bytes are transferred from the index register (R), or the accumulator if R=000.

The write-address command must be issued once to establish the desired addresses in the output buffer register. Any number of write-control-and-data commands, described next, can then be issued without having to issue another write address command. Another write address command must be issued only when the address(es) must be changed.

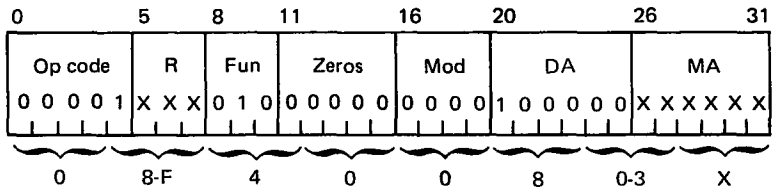
Write Control and Data



This command stores the control and data bytes into the last two bytes of the output buffer register in the 2790 Control. These two bytes are transferred from the index register (R), or the accumulator if R=000.

After these bytes are loaded into the output buffer register, the 2790 Control transmits to the 2790 system an active frame cycle that contains the information in the output buffer.

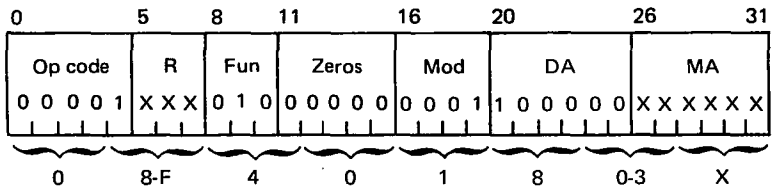
Read Address



This command stores the first two bytes of the input buffer register (area station address and device address) into the index register (R), or the accumulator if R=000.

These addresses are obtained by the program so that the contents of the incoming frame can be compared with the contents of the outgoing frame for validity and for future use in outgoing frames. The addresses may also be placed by the program in the next active outgoing frame that is generated.

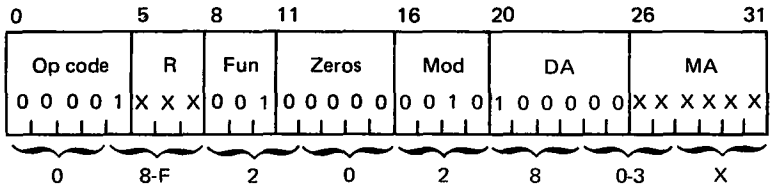
Read Control and Data



This command stores the last two bytes of the input buffer register (control and data bytes) into the index register (R), or the accumulator if R=000.

The control and data bytes should be checked for valid responses and data, respectively, every time a frame is received from the area station determined by a read address command (described previously). The results of this check may also determine the control and/or data that should be placed in the next outgoing frame. (See Figure 13-5 for valid 2790 Control commands and responses.)

Set Diagnostic Mode

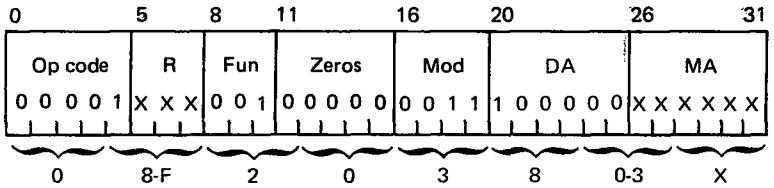


This command places the 2790 Control function in one of four states, depending upon the value of bits 14 and 15 in the index register (R), or the accumulator if R=000.

The four possible states set by this command are:

Bits		*State*
14	*15*	
0	1	Diagnostic mode. The 2790 Control is set in diagnostic mode of operation for use with the set-input-buffer command described later. ISW bit 5 is also set on to indicate that the diagnostic mode is effective. While in the diagnostic mode, the 2790 Control constantly checks for the presence of sync characters in bytes 5 to 29 of a frame cycle. Any missing sync bytes are indicated by setting on bit 6 in the ISW.
1	0	Loop bypass. The 2790 loop is bypassed. Output from the output buffer register is connected directly to the input of the input buffer register.
1	1	Diagnostic and loop bypass. Output frames from the 2790 Control enter directly into the input buffer register and are checked for sync characters in bytes 5 to 29.
0	0	Normal operation. This code resets the above three states and removes the 2790 Control from the diagnostic and/or bypass modes of operation.

Set Input Buffer Control



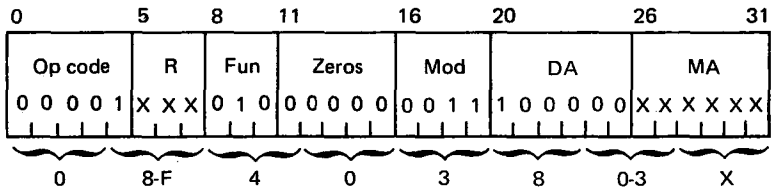
This command loads a sync character (hexadecimal 47) or a character containing all 1's into the input buffer register, depending on the setting of bit 15 in the index register (R), or the accumulator if R=000. A set-diagnostic-mode command must set the 2790 Control function in diagnostic mode before executing a set-input-buffer-control command.

Bit 15 in the R register can be set as follows:

Bit 15 = 0	An interruption request is presented if an inactive frame cycle is received (bytes 2 to 5 are sync characters).
Bit 15 = 1	The input shift register shifts logical 1's into the input buffer register. When all positions in the register contain 1's, an interruption request is presented to the processor module. Since the 2790 Control is in diagnostic mode, this "forced" shifting checks the input circuits without using the output portion of the function.

A read-address or read-control-and-data command would normally follow the set-input-buffer-control command when the interrupt is requested by the 2790 Control.

Read ISW



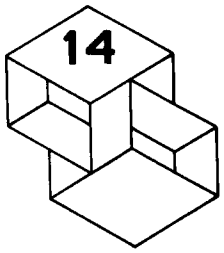
This command stores the 16-bit interrupt status word (ISW) into the index register (R), or the accumulator if R=000.

Interrupt Status Word (ISW): The 16 bits in the interrupt status word are set to indicate operating status of the 2790 Control function and to detect errors that occur after a write-control-and-data command is completed, but while the device is still busy. The ISW bits are reset on the first selection of the 2790 Control function after the interruption request has been accepted by the processor module.

Significant bits in the 2790 Control ISW and their meanings are:

Significant Bits	*Meaning*
3	Frame parity error. One of the four bytes of an active frame received in the input buffer had a parity error.
4	No input detected. The 2790 Control input circuits did not receive an input pulse from the loop within 1.024 ms.
5	Diagnostic mode. The 2790 Control is in the diagnostic mode of operation. This is not an error.
6	No sync bytes. A byte other than a sync byte was detected in bytes 5 to 29 of a frame cycle. This is valid only when the 2790 Control is in diagnostic mode, and is not an error.
7	Bypass mode. The 2790 Control function is in the bypass mode of operation. This is not an error.
12	Device busy. The 2790 Control is transmitting an active frame and is waiting for the interrupt from the frame returning in the input buffer.
13	Device end. The 2790 Control has completed the requested operation.

CHAPTER CONTENTS



The 5013 Digital I/O Module Model A1 groups the following functions within one physical enclosure:

- As many as 128 digital input points
- As many as 64 digital output points
- One 2790 Control for an IBM 2790 Data Communication System

Any combination of the above functions can be installed in a single digital I/O module. Programming support, however, is provided for only four 2790 Controls in a System/7 configuration (this includes any 2790 Controls in 5012 Multifunction Modules). With the single exception that *all* 2790 Controls must be located within the *same* 5026 Enclosure, the digital I/O modules can be located in any or all of the I/O module positions within a 5026 Enclosure (location 00 in the 5026 is the processor module location).

FUNCTIONAL DESCRIPTION

Digital Input

As many as 128 isolated and/or non-isolated digital input points can be accommodated, in eight groups of 16 points each. Isolated two-terminal inputs can be activated by either contact or voltage sources provided by the user's equipment.

The first two digital input groups can each be provided with a special feature that permits them to interrupt the System/7 processor. A 16-point group is compared with a 16-bit reference register, and interruptions can be initiated on the basis of either an equal or an unequal comparison.

A non-isolated digital input group provides 16 latching or non-latching points of contact sensing capability. A non-isolated group cannot, however, have the process interrupt feature.

Digital Output

As many as 64 isolated and/or non-isolated digital output points can be accommodated, in four groups of 16 points each. Each isolated group can use either of the two optional digital output circuits available (medium-power group or contact group). These circuits are described in more detail in the *System/7 Installation Manual–Physical Planning,* Order No. GA34-0004.

A medium-power non-isolated digital output group provides 16 solid-state switches for switching up to 52.8V dc and 450 ma of user-supplied power.

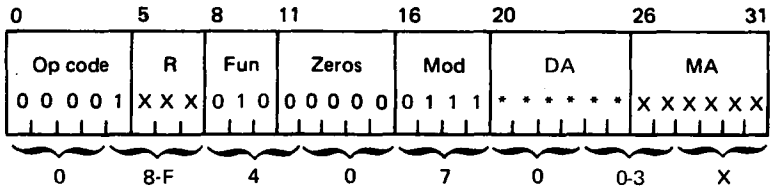
2790 Control

A single loop of the IBM 2790 Data Communication System can be attached to the digital I/O module. Such an attachment allows the System/7 to act as system controller to the 2790 devices (2791/2793 Area Stations, 2792 Remote Communications Controller, and 2795/2796/2797 Data Entry Units). Data enters the 2790 system from a card reader, badge reader, pulse counter, or key entry unit. Output data to the 2790 system is sent to 2791 indicator lights or to a printer attached to either a 2791 or 2793 Area Station.

COMMON I/O COMMANDS

Three I/O commands are common to all functions in the digital I/O module: read DSW, read ID, and read-ID-extension. In these three commands, the contents of the device address field are ignored, but the module address is determined by the physical location of the digital I/O module in the 5026 Enclosure. The read ID and read-ID-extension commands are described earlier in Chapter 4 under "Input/Output Instruction," since these two commands apply to all I/O modules.

Read DSW



The read DSW command stores the 16-bit device status word (DSW) associated with the digital I/O module into the index register (R), or the accumulator if R=000. Execution of this command resets the bits in the DSW.

Device Status Word (DSW): The 16 bits in the device status word are set as a result of detected error conditions occurring during the execution of an immediate command to the I/O module. The operating program is notified of these error conditions by returning condition code 1 to the processor. If other commands are attempted before resetting the DSW bits, condition code 1 continues to be returned to the program.

The significant bits in the DSW and their meanings are:

Significant Bits	*Meaning*
1	Command reject. The digital I/O module cannot execute the command. (For example, an interruption-causing command issued to the device when it is disabled for interruptions.)
3	Open circuit. A digital input circuit breaker is open when the digital input subaddress was selected.
14	Data check. The digital I/O module has detected a parity error involving data. The operation that caused this condition usually can be retried successfully if the error condition is intermittent.
15	Invalid device address. An immediate command was issued to a device that is not installed in the digital I/O module.

MODULE DEVICE ADDRESSES

I/O commands to the digital I/O module use the six-bit device address field (bits 20 to 25 of the PIO instruction) to specify the particular function being addressed. The device address bit assignments for each function are:

Device Address	*Function*
000000	Digital input group 0 with or without process interrupt feature
000100	Digital input group 1 with or without process interrupt feature
001000	Digital input group 2
001100	Digital input group 3
010000	Digital input group 4
010100	Digital input group 5
011000	Digital input group 6
011100	Digital input group 7
100000	2790 Control
110000	Digital output group 0
110100	Digital output group 1
111000	Digital output group 2
111100	Digital output group 3
XXXXYY	The last two bits (YY) are the quadrant. When YY is not 00, the address is assigned to a custom attachment.

The prepare I/O command is used as described in Chapter 4 under "Input/Output Instruction." Only three device addresses are used with the prepare I/O command in order to prepare an input source to interrupt. The three device addresses and their functions are:

Device Address	*Function*
000000	Prepare digital input group 0 with process interrupt feature
000100	Prepare digital input group 1 with process interrupt feature
100000	Prepare 2790 Control

DIGITAL INPUT CONTROL

The digital input control function of the digital I/O module can accommodate as many as 128 isolated and/or non-isolated digital input points, in eight groups of 16 points each. These points are two-terminal inputs that can be activated by either contact or voltage sources provided by the user's equipment. A non-isolated digital input group provides 16 latching or non-latching points of contact sensing capability.

A digital input contact point has a 1 value in its corresponding bit when the contact is closed, and a 0 value when the contact is open.

An isolated digital input voltage point has a 1 value in its corresponding bit when the input is from +2V to +52.8V, and a 0 value when the input is from −52.8V to +0.6V. A non-isolated digital input voltage point has the opposite polarity sensing; a 0 value from +2V to +52.8V, and a 1 value from −52.8V to +6.0V. An input between +0.6V and +2V, in either case, produces an indeterminate result.

Each group of digital input points can be program controlled to provide either latched or unlatched input. In the latched mode of operation, each bit that is set on in the group remains in that state (regardless of any further change in input signal or contact operation) until the group is reset or unlatched. A bit is not reset if the input signal remains active (voltage present or switch closed) throughout the execution of the read/reset digital input register command. In the unlatched mode of operation, each bit in the group is set on or off dynamically as determined by the input fluctuations.

Digital input circuits are compatible with digital output circuits, which permits multiplexing of digital output points with the user's contacts for digital input sensing applications. If all of the points are isolated, either positive or negative voltage sources can be used through proper connection of input sources to the input terminals. This circuit compatibility also permits direct connection of digital input points to digital output points for testing of the input/output circuits.

Process Interrupt Feature

A Process Interrupt feature can be obtained for one or both of the first two digital input groups (group 0 and/or group 1). Each of the two groups is a separate interrupting source and, therefore, can be assigned its own priority level and sublevel by separate prepare I/O commands. Group 0 is installed when only one group is ordered. The Process Interrupt feature can be attached only to isolated groups.

With the Process Interrupt feature, a 16-point group is compared with a 16-bit reference register (which can be set to any value by the program), and an interruption request is presented on the basis of either an equal or an unequal bit-by-bit comparison. The choice of comparison is under program control and can be changed at any time. When a group has been conditioned to perform a compare operation, the program cannot read the digital input register for that group until an interruption request is presented. After the request is presented, the group will no longer be conditioned to perform a compare operation. The group can then be used the same as a noninterrupting digital input group.

The Process Interrupt feature can be used with groups 0 and/or 1 when they are operating in either the latched or unlatched mode of operation. In the latched mode, each bit that is set on in the group remains in that state (regardless of any further change in input signal or contact operation) until the group is reset or unlatched. All bits in the group are reset, even if the input signal remains active throughout the execution of the read/reset digital input register command. In order to again set a bit in the digital input register, a positive transition must occur; for example, the input point must have gone to a 0 value (+0.6V to −52.8V) and then back to a 1 value (+2V to +52.8V) after the digital input register was reset. When changing modes, latched to unlatched or unlatched to latched, and the user's input value is in the logical 1 state, the digital input register indicates a logical 0 until such time as another positive transition takes place.

In the unlatched mode of operation, each bit in the group is set on or off dynamically as determined by the input signal fluctuations. The output of the digital input register follows the user's input value; for example, the user's value at a logical 0 results in a logical 0 output from the digital input register.

Exception: The output of the digital input register will *not* follow the user's input value in unlatched mode whenever a switch from latched mode to unlatched mode takes place and the user's input value remains in the logical 1 state. Either of two methods can be used to restore the digital input register to again reflect the user's input values.

1. Program the following sequence:
   −Issue a set-test-signal command (bit 15=0).
   −Establish a 3-microsecond delay.
   −Issue a read/reset-digital-input-register command.
2. Ensure that the user's input value returns to a logical 0 after the completion of latched mode operation.

Even though the Process Interrupt feature is installed, it can be disabled and enabled under program control at any time. Program control of the Process Interrupt feature is provided by the set-digital-input-interrupt-control command.

I/O Commands

The prepare I/O command is used as described in Chapter 4 under "Input/Output Instruction" and in this chapter under "Module Device Addresses." This command applies only to groups 0 and 1 because only these two groups can have the Process Interrupt feature. The prepare I/O device address field must be 000000 for group 0 or 000100 for group 1.

Data transfers between the digital input control and the processor module are accomplished by immediate read and immediate write commands. The setting of the modifier field bits in the command further defines the operation to be performed.

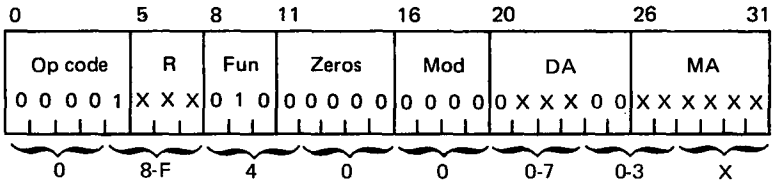
All read and write I/O commands addressed to the digital input control must use one of the following device addresses in order to address the corresponding digital input group separately:

Device Address	*Digital Input Group*
000000	0
000100	1
001000	2
001100	3
010000	4
010100	5
011000	6
011100	7

The module address in the I/O command varies depending upon the physical location of the digital I/O module in the 5026 Enclosure.

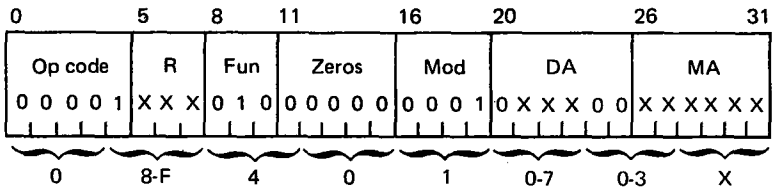
The normal programming routines for digital input are shown in Figures F-3 and F-4 of "Appendix F."

Read Digital Input Register



This command stores the 16 bits of input data from a group (identified by the device address field) into the index register (R), or the accumulator if R=000.

Read/Reset Digital Input Register



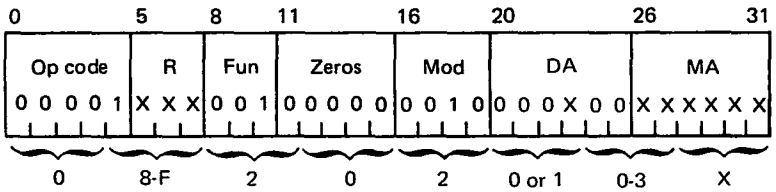
This command stores the 16 bits of input data from a group (identified by the device address field) into the index register (R), or the accumulator if R=000.

If a group is non-isolated and operating in latched mode, its digital input register is reset to 0 only if its input has returned to a 0 value. If the input is at a 1 level, the register position remains set to 1.

If the group is isolated, operating in latched mode, and the input is not active, its digital input register is then reset to 0.

If a process interrupt group is operating in latched mode, its digital input register is then reset to a 0 value (even if the input is still active).

Set Digital Input Reference Register

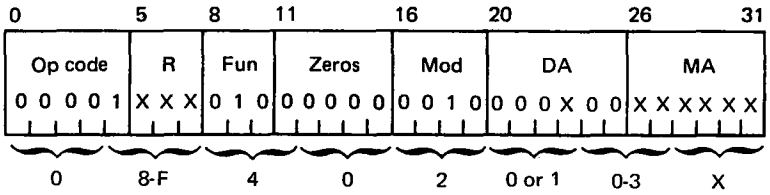


This command stores a 16-bit value in the reference register associated with group 0 or group 1. The choice is determined by the device address field in the command: 000000 for group 0 or 000100 for group 1.

The value set into the reference register is obtained from the index register (R), or the accumulator if R=000.

The reference registers are part of the Process Interrupt feature for groups 0 and 1, and are used for comparison with the 16-point group input register. Interruption requests can be presented for either an equal or an unequal comparison; the choice is under control of the program. (Refer to "Set Digital Input Interrupt Control" later in this chapter.)

Read Digital Input Reference Register

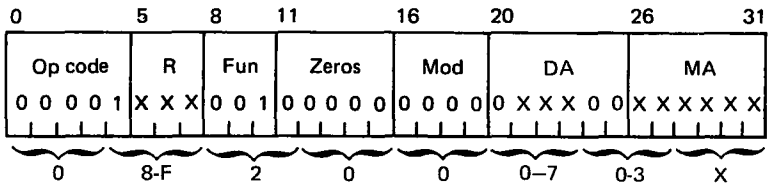


This command stores the contents of the 16-bit reference register associated with group 0 or group 1. The choice is determined by the device address field in the command: 000000 for group 0 or 000100 for group 1.

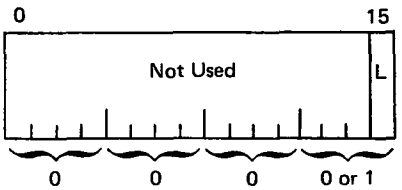
The reference register contents are stored into the index register (R), or the accumulator if R=000.

The reference registers are part of the Process Interrupt feature for groups 0 and 1, and are used for comparison with the 16-point group input register. Interruption requests can be presented for either an equal or an unequal comparison; the choice is under control of the program. (Refer to "Set Digital Input Interrupt Control" later in this chapter.)

Set Digital Input Latch Control



This command sends control information to the digital input group identified by the device address field. The control information is indicated by the value of bit 15 of the index register (R), or the accumulator if R=000, and has the following format:



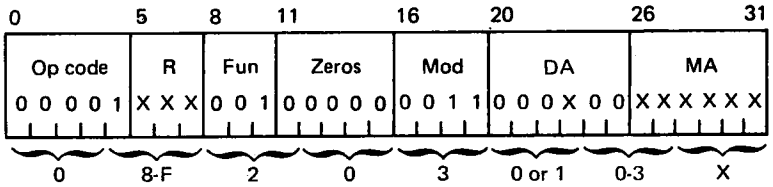
Control information in bit 15 can be directed to any of the eight possible digital input groups (groups 0 to 7) in a digital I/O module. The setting of bit 15 determines whether the selected group is to operate in the latched or unlatched mode. Bit 15 set to a 1 value specifies the latched mode, and a 0 value specifies the unlatched mode.

When the Process Interrupt feature is installed, changing modes (latched to unlatched or unlatched to latched) resets the digital input register to the all 0's state. The register will remain all 0's until the next positive transition of the user's input signal (logical 0 to 1). Setting bit 15=0 also resets the digital input register to the all 0's state.

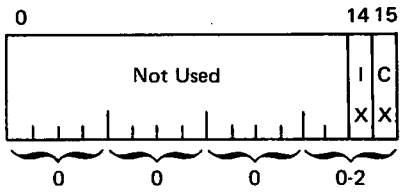
Note. If while in the latched mode of operation, the user's signal has remained in the logical 1 state when unlatched mode of operation is selected, the group must be reinitiated before normal unlatched mode operation takes place. Either of two methods can be used to restore the digital input register to again reflect the user's input values.

1. Program the following sequence:
   –Issue a set-test-signal command (bit 15=0).
   –Establish a 3-microsecond delay.
   –Issue a read/reset-digital-input-register command.
2. Ensure that the user's input value returns to a logical 0 after the completion of latched mode operation.

Set Digital Input Interrupt Control



This command sends control information to the Process Interrupt feature of digital input group 0 or 1. The choice is determined by the device address field in the command: 000000 for group 0 or 000100 for group 1. The control information is indicated by the value of bits 14 and 15 of the index register (R), or the accumulator if R=000, and has the following format:

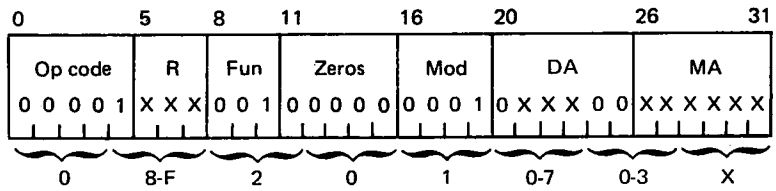


Forced interruption control is provided by the setting of bit 14. If bit 14 is on, a process interrupt is forced for the selected group (group 0 or 1). The setting of bit 15 determines if interruption requests are to be presented for an equal or an unequal comparison between the selected digital input group register and the reference register. When bit 15 is set on, an interruption request is presented to the system for an equal comparison. When bit 15 is set off, the interruption request occurs for an unequal comparison.

The set-digital-input-interrupt-control command turns on the device-busy bit in the digital input ISW. The addressed digital input group remains busy to all subsequent commands (except another set-digital-input-interrupt-control or a read ISW command) until an interruption request has been presented by the group.

This command applies to digital input groups 0 and 1 only if they have the Process Interrupt feature installed. If this feature is not installed, the command does not apply to the Digital Input feature.

Set Test Signal



In unlatched mode of operation, this command sets a digital input register to a predetermined value, regardless of the user's input value. Selection of the desired digital input group is made by the device address field contents in the command to the addressed digital I/O module.

The value to be set into the digital input register is determined by the setting of bit 15 in the index register (R), or the accumulator if R=000. If bit 15=0, the 16-bit digital input register is set to all 0's. If bit 15=1, the digital input register is set to all 1's. A 3-microsecond delay must precede the issuance of a read/reset or read-digital-input-register command to obtain the test data. A read/reset-digital-input-register command must be issued to restore the digital input circuits to their original status.

Digital Input Groups Without Process Interrupt Feature: In latched mode, operation is the same as in unlatched mode except under these two conditions.

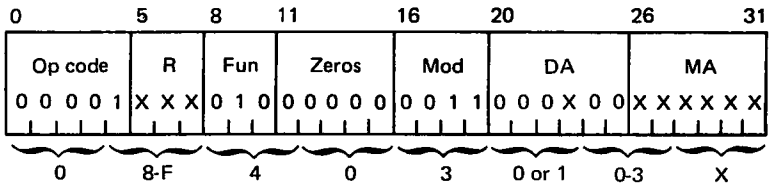
1. Whenever the following sequence of instructions is issued, two successive read/reset digital-input-register commands must be issued to restore the digital input circuits to their original status:
   –Set-digital-input-latch-control (bit 15=1).
   –Set-test-signal (bit 15=1).
   –3-microsecond delay.
2. The following sequence of instructions results in reading the user's input value instead of reading all 0's:
   –Set-digital-input-latch-control (bit 15=1).
   –Set-test-signal (bit 15=0).
   –3-microsecond delay.

Digital Input Groups With Process Interrupt Feature: In latched mode, operation is the same as unlatched mode except under the following conditions:

When any of the user's input signals are in the logical 1 state at the same time that a set-digital-input-latch-control command is issued (bit 15=1) followed by a set-test-signal-to-1's instruction, a read-digital-input-register command or a read/reset-digital-input-register command will read 1's for all bits whose inputs were at the logical 0-state.

Note. If the sequence of commands is such that a set-test-signal command (bit 15=0 or bit 15=1) is followed by a mode change (latched to unlatched or unlatched to latched), the digital input register is set to the all 0's state.

Read ISW



This command stores a 16-bit interrupt status word (ISW) into the index register (R), or the accumulator if R=000.

Interrupt Status Word (ISW): No ISW is associated with the digital input function unless the process interrupt feature is installed. For each process interrupt feature (maximum of two), there is a separate ISW, one for each group with interruption capability.

The setting of ISW bits (except the busy bit) is indicated to the operating program by an interruption request. Interruption requests are presented because of errors detected in digital input operations, as well as for the occurrence of a process interrupt.

A read ISW command also resets the addressed ISW. The significant bits in the ISW and their meanings are:

Significant Bits	*Meaning*
3	Open circuit. A digital input circuit breaker has opened during the time that the corresponding digital input group is busy. This condition results in an interruption request.
12	Device busy. Digital input group 0 or 1 was addressed and that group is in an interruption-enabled state. (See "Set Digital Input Interrupt Control" previously described in this chapter.) The resulting interruption on equal or unequal comparison turns off this bit and turns on the device-end bit.
13	Device-end. A successful compare on equal or unequal has occurred for group 0 or 1. This causes an interruption request and disables interruptions for that group.

DIGITAL OUTPUT CONTROL

The Digital Output Control feature of the digital I/O module can accommodate as many as 64 isolated and/or non-isolated digital output points, in four groups of 16 points each. Each group can be configured to use one of the three optional digital output circuit types available (low-power group, medium-power group, and contact group). A non-isolated digital output group provides 16 medium-power, solid-state switches for switching up to 52.8V dc and 450 ma of user-supplied power.

The digital output points can be operated by either of two methods: (1) data can be sent directly to a selected digital output group register which, in turn, operates the output points; or (2) data can be sent to the holding register and then read back and checked for accuracy before transferring it to one of four digital output group registers.

Digital output circuits are compatible with digital input circuits, which permits multiplexing of digital output points with the user's contacts for digital input sensing applications. Either positive or negative voltage sources can be used through proper selection of the input terminals when the digital output medium-power group or digital output contact group is used. This circuit compatibility also permits direct connection of digital input points to digital output points for testing of the input/output circuits.

I/O Commands

Data transfers between the digital output function and the processor are accomplished by immediate read and immediate write commands. The setting of the modifier field bits in the command further defines the operation to be performed.

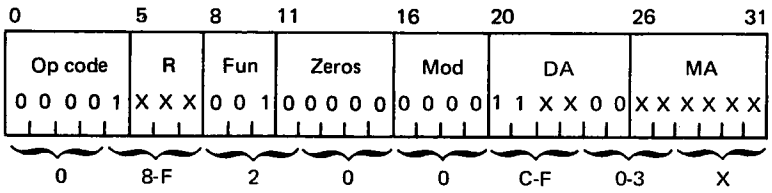
All read and write I/O commands addressed to the digital output control must use one of the following device addresses in order to address the corresponding digital output group separately:

Device Address	*Digital Output Group*
110000	0
110100	1
111000	2
111100	3

The module address in the I/O command varies depending upon the physical location of the digital I/O module in the 5026 Enclosure.

The normal digital output programming routine is shown in Figure F-5 of "Appendix F."

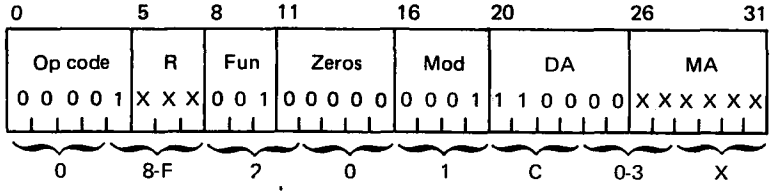
Write Digital Output Group Register



This command stores 16 bits of output data from the index register (R), or the accumulator if R=000, into the digital output group register identified by the device address field.

The status of the 16-bit digital output register is reproduced at the corresponding terminals for its associated 16-point group. A 1-bit in the register produces a closed or on condition at its corresponding output point; a 0-bit in the register produces an open or off condition at its corresponding output point.

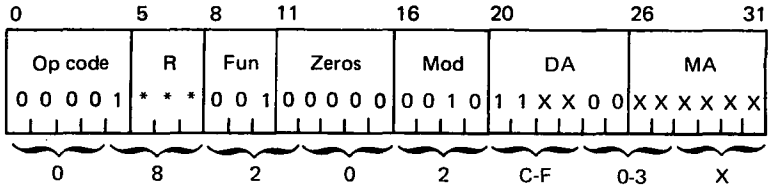
Write Digital Output Holding Register



This command stores 16 bits of output data from the index register (R), or the accumulator if R=000, into the holding register.

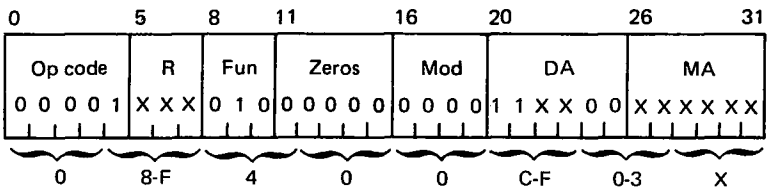
The holding register is a buffer that allows the data to be verified by the program prior to transferring it to one of four output registers (and to the corresponding terminals).

Set Digital Output Group Register



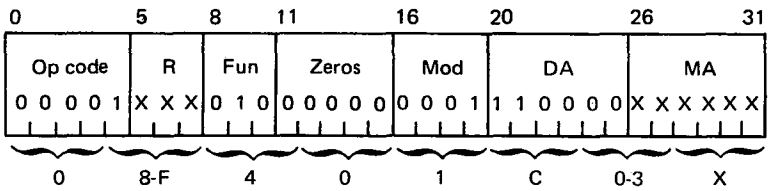
This command transfers the 16 bits of output data in the holding register to the digital output group register identified by the device address field. The R field is ignored.

Read Digital Output Group Register



This command stores the 16 bits of output data from a group register (identified by the device address field) into the index register (R), or the accumulator if R=000.

Read Digital Output Holding Register



This command stores the 16 bits of output data from the holding register into the index register (R), or the accumulator if R=000.

2790 CONTROL

The 2790 Control feature of the digital I/O module attaches a single loop of the IBM 2790 Data Communication System to the System/7 as shown in Figure 14-1. This feature provides the user with a flexible plant communication and data collection system.

The 2792 Remote Communications Controller attaches a remote loop to the 2790 single loop as shown in Figure 14-1. This capability extends the System/7's locally controlled 2790 Data Communications System to remote locations via a full duplex (with four-wire termination) leased common-carrier line. The transmission rate is 2400 bits per second.

The 2790 system provides for rapid transfer of digital information from data entry units at various plant locations to area stations at key locations and on to a system controller. The System/7 is the system controller for a 2790 system attached to the 2790 Control in the 5013 Digital I/O Module. Data enters the 2790 system from a card reader, badge reader, pulse counter or key entry unit. Output data to the 2790 system is sent to 2791 indicator lights, to the 2798 display and guidance panel, or to a printer attached to either a 2791 or 2793 Area Station.

The 2790 Data Communication System is connected by a two-wire, high-speed loop capable of operating at about 500,000 bits per second (about 900 characters per second) partitioned over the active terminals on the loop. This loop is a serial connection that starts and ends at the 2790 Control feature in the System/7.

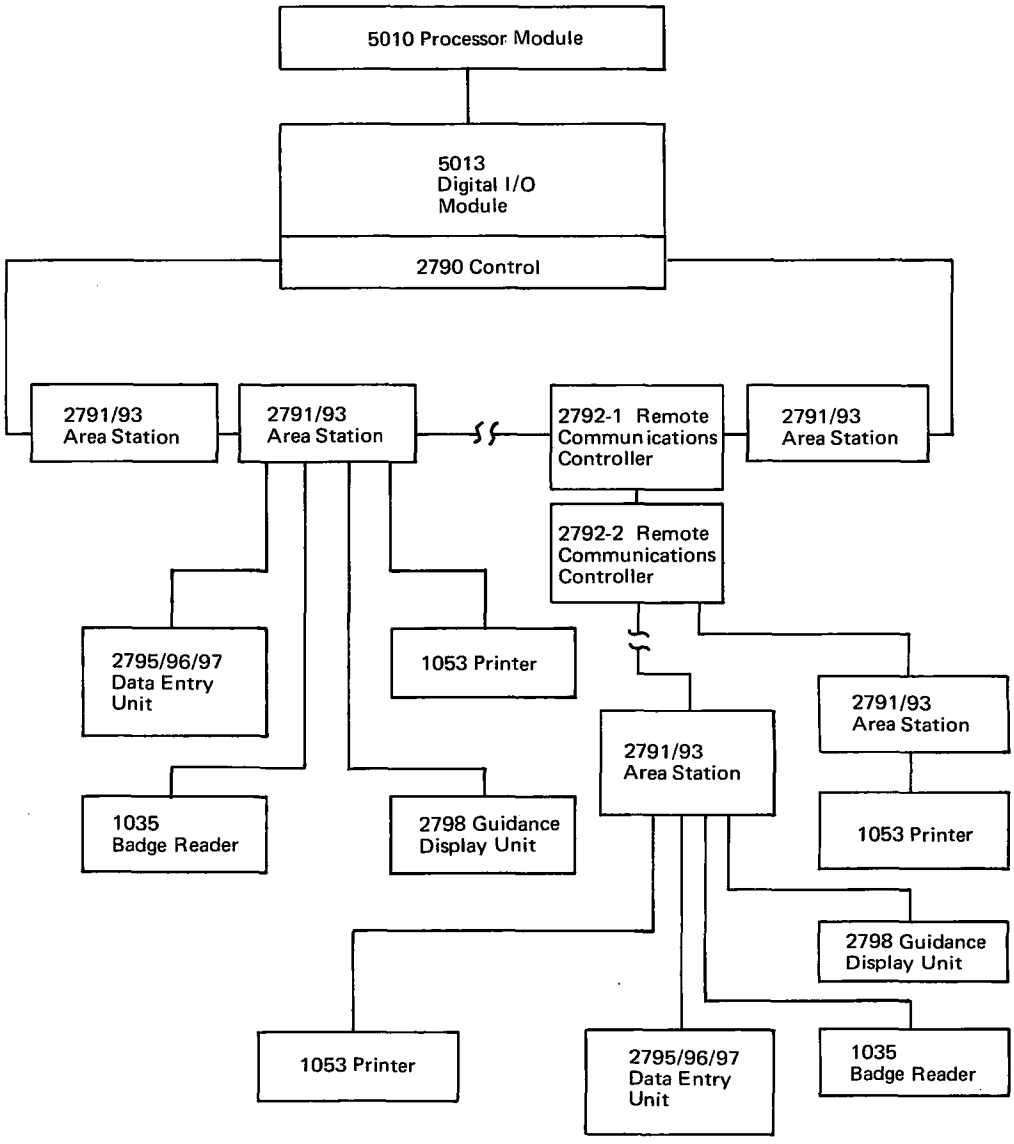


Figure 14-1. 2790 Data Communication System attached to System/7

Multiple area stations (2791 and/or 2793) attach to the transmission loop by inserting a one-byte shift register in series with the line. This shift register (Figure 14-2) enables the area station to examine and modify a full byte of information before passing it on to the next area station in the loop. Multiple 2795/2796/2797 Data Entry Units, 1035 Badge Readers, 2798 Guidance Display Units, and 1053 Printers can be connected to each area station. The maximum distance between any two area stations on the loop, and between the first/last area station and the 2790 Control, is governed by the type of wire used. See Appendix B in the manual *IBM 2790 Data Communication System: Installation Manual–Physical Planning,* Order No. GA27-3017. Remote locations may be serviced via a remote loop attached through a 2792 pair.

Programming and storage requirements to support the 2790 system limit the actual configuration of area stations and associated input and output units that may be attached to the loop. These limits are specified in manuals that describe programming support for the particular system using the programs. Programming support is provided for up to four 2790 Control Features in a System/7 configuration.

For more detailed information on the capabilities, components, features, and applications of the 2790 Data Communication System, refer to the manuals *IBM 2790 Data Communication System Component Description,* Order No. GA27-3015, and *IBM 2790 System Summary,* Order No. GA27-3016.

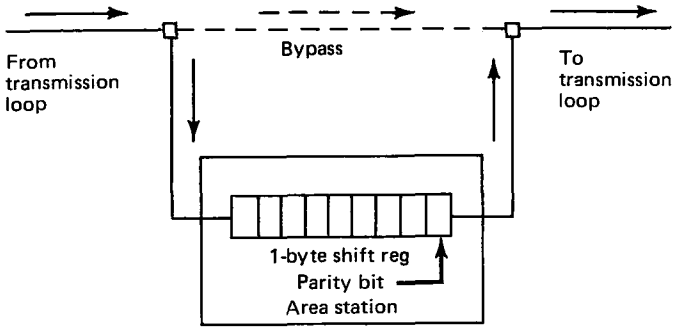


Figure 14-2. One-byte shift register

Data Transmission

Information is transmitted over the loop by data bits, represented as bipolar pulses at approximately 500 kHz, as long as the 2790 Control in the 5013 Digital I/O Module has power. All communication between the System/7 and the area stations is accomplished by a sequence of five information bytes, called a frame, shown in Figure 14-3. Byte 0 (the start byte) is generated by the 2790 Control; the remaining four bytes (area station address, device address, control, and data) are generated by the program in the processor module. Figure 14-3 shows that 25 synchronization bytes separate each five-byte active portion in a frame cycle, for a total of 30 bytes. These 30 bytes make up what is called an active frame. All synchronization bytes are generated by the 2790 Control.

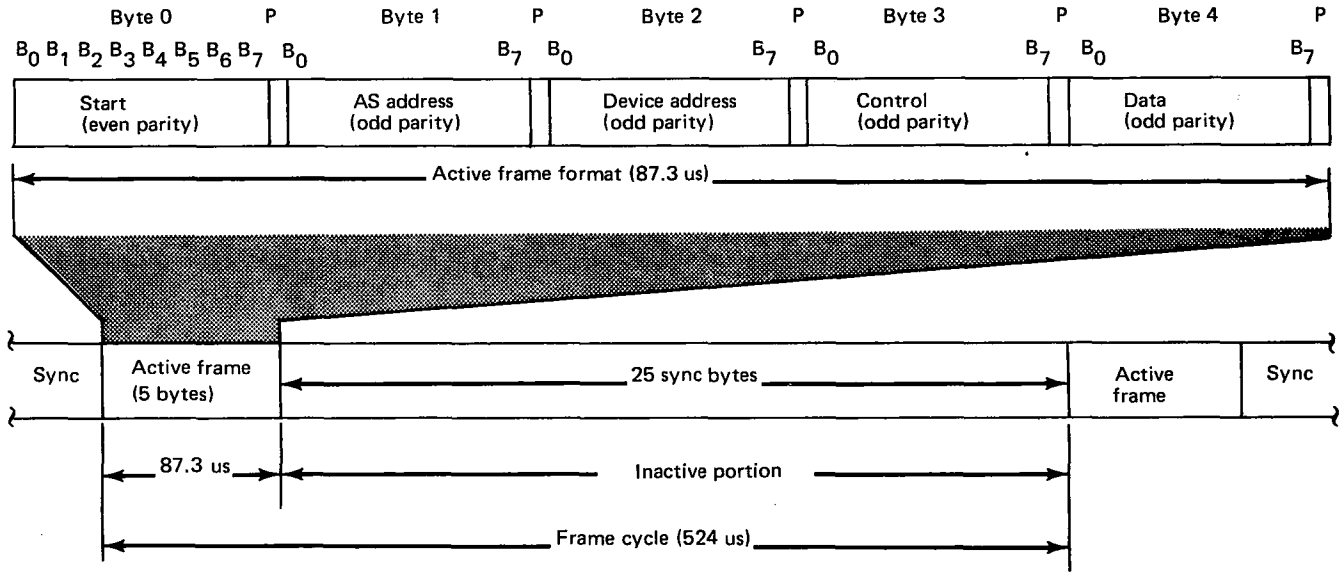


Figure 14-3. Data transmission format

Frames are transmitted by the 2790 Control at intervals of 524 microseconds. Active frames can be separated by one or more inactive frames, which consist of a start byte and 29 synchronization bytes. If the controlling program does not specify a new active frame, an inactive frame is sent out by the 2790 Control. This separation is used primarily to allow time to process input frames and to prepare output frames. The frame fields are summarized in Figure 14-4.

When multiple 2790 Controls (up to 4) are installed in a System/7 configuration, the output frames transmitted by each 2790 Control are offset by 3 bytes from the output frames transmitted by any other 2790 Control. In other words, the first 2790 Control starts transmitting output frames. When byte 3 of its first output frame is sent, then the second 2790 Control starts transmitting. When byte 3 of its first output frame is sent, then any third 2790 Control starts transmitting. When byte 3 of its first output frame is sent, then any fourth 2790 Control starts transmitting. Once begun, the transmission of output frames from all 2790 Controls is continuous.

Note: When active frames are transmitted too rapidly to the same device, critical timing conditions may occur. These conditions are described later in this chapter under "Critical Timing Conditions."

Field	Byte	Hex	Bit pattern 0 1 2 3 4 5 6 7 P
Start	0	39	0 0 1 1 1 0 0 1 0
AS address	1		
Discrete AS		80-FF	1 A A A A A A A -
All AS		09	0 0 0 0 1 0 0 1 1
Any AS		11	0 0 0 1 0 0 0 1 1
Software idle		18	0 0 0 1 1 0 0 0 1
Device address	2		Select Adapter
Diagnostics		00	0 0 0 0 0 0 0 0 -
Pulse counter		01-3F	0 0 A A A A A A
Data entry unit		C0-DF	1 1 0 A A A A A -
Local output			
1053 Printer		40	0 1 0 0 0 0 0 0 0
Area station display		80	1 0 0 0 0 0 0 0 0
Local input			
Badge reader		80	1 0 0 0 0 0 0 0 0
Card reader		84	1 0 0 0 0 1 0 0 1
Keyboard		88	1 0 0 0 1 0 0 0 1
OEM		8C	1 0 0 0 1 1 0 0 0
1035 Badge Reader			1 0 0 0 0 0 A A -
Remote X		81	1 0 0 0 0 0 0 1 1
Remote Y		82	1 0 0 0 0 0 1 0 1
Remote Z		83	1 0 0 0 0 0 1 1 0
Guidance Display Unit		90-9F	1 0 0 1 A A A A -
Control (Refer to Figure 14-5 for all command/response bit patterns)	3		AS Response digit / Command digit U V W W Y Y Z Z Op code Modifier AS defined Status bit Data mode Restore Ack
Data, status, or guidance	4		D D D D D D D D -
Sync (hardware idle)	5-29	47	0 1 0 0 0 1 1 1 1

"A" denotes an address bit

"AS" denotes area station

Figure 14-4. Frame field code structure

The functions of the five bytes in an active frame are described in the following paragraphs.

Start

The start byte, which contains hexadecimal 39, signals the start of a frame cycle. The end of an active frame is indicated by the selected area station reaching a byte count of five. The start byte, generated by the 2790 Control, is the only byte that requires even parity.

Area Station Address

Coded information in this byte can be used to select a unique area station, any area station, or all area stations.

There are 128 unique area station addresses, which must be indicated by codes within the range of hexadecimal 80 to hexadecimal FF. All address codes outside this range are invalid. Because of the characteristics of the system, it is recommended that area station addresses be assigned sequentially starting with hexadecimal 80. This does not, however, restrict the physical placement of area stations on the loop.

An any-area-station address is designated by a code of hexadecimal 11. Whenever an area station requires service from the program, it searches for a frame having this code. Upon finding one, the area station captures the frame by inserting its own address in place of the any-area-station address in this byte.

An all-area-stations address is designated by a code of hexadecimal 09. Every area station executes the instruction specified in the control byte, if it can, without modifying the area station address byte.

A software idle is designated by a code of hexadecimal 18. This code is used by the 2790 Control programming to reserve a specific frame. No action is taken by the area station.

Device Address

A frame addressed to an area station is also addressed to a particular device (such as a 2795 Data Entry Unit) on a specific adapter connected to that area station. Bits 0 and 1 of the device address byte identify the particular adapter in the area station, or determine that this byte is a part of diagnostic controls. Bits 2 to 7 identify the device attached to the addressed adapter. The diagnostic controls are normally addressed by hexadecimal 00. However, a diagnostic command is executed regardless of the contents of this byte. If an area station adapter is selected, or an area station captures an any-address frame, the area station inserts its own address and the address of a particular device into the frame.

Control

The control byte is divided into two hexadecimal digits. The low-order digit (bits 4 to 7) contains the command issued by the 2790 Control. The command digit is further subdivided into an operation code (bits 6 and 7) and a modifier (bits 4 and 5).

The high-order digit (bits 0 to 3) contains the response from an area station. An addressed area station always responds to a command that is valid for that particular area station. Bit 0 of the response digit is the restore acknowledge bit and is nearly always set to 0. Bit 1 is the data mode bit. It is set on when the area station decodes one of the diagnostic commands, or when the addressed adapter is in the data mode and decodes a valid operation code. Bits 2 and 3 are status bits, and are defined by the area station to reflect the status of an operation in the area station adapter.

The control byte codes permit two-way communication between the area stations and the system controller for every step during the various operations.

Data

The data byte contains either data or status information, depending upon the contents of the control byte.

A data transfer caused by a read command always involves receiving the data byte twice from the area station, once in each of two frames. Thus, input data can be checked for accuracy without special checking logic in the area station.

Data transferred as a result of a write command is also checked for accuracy by transmitting it to an area station and then receiving the data back from the area station. If the data received is the same as the data transmitted, the area station is then commanded to print the data or turn on a specified indicator light.

Commands and Responses

The control byte of each transmission frame contains coding and responses to all commands. All commands, which are generated by the System/7 program, are sent from the 2790 Control to an area station, whereas all responses are returned from an area station to the 2790 Control. Thus, a control byte transmitted from the 2790 Control should contain a response when that command is received in the 2790 Control.

Each command and response is represented by two hexadecimal digits corresponding to the coding within the eight-bit control byte. The first hexadecimal digit (four high-order bits of the control byte) is called the area station response digit. The second hexadecimal digit (four low-order bits of the control byte) is called the command digit. This command digit is further divided into a two-bit modifier and a two-bit op code as shown in Figure 14-5.

Bit 0 of the response field is turned on to indicate that a frame is being sent by the 2792 to the remote loop and a response is pending.

Command frames are easily distinguished from response frames because the response digit is zero for all commands, but nonzero for all responses as shown in Figure 14-5. There are several commands in each of the three major command categories: read, write, and control. In addition, there is at least one valid response for each of the commands.

Note: When power is initially applied to the area stations, they are in the bypass mode of operation. (This mode is explained later in this chapter.) The controlling program must issue control commands to restore discrete area stations, or a single control command addressed so as to restore all area stations. The restore operation inserts the area station(s) into the transmission loop.

Code name	Hex	Bit pattern	
		Response 0 1 2 3	Command 4 5 6 7
Read commands (from System/7)			
Read	06	0 0 0 0	0 1 1 0
Read null	02	0 0 0 0	0 0 1 0
Read data	0A	0 0 0 0	1 0 1 0
Read end	0E	0 0 0 0	1 1 1 0
Read responses (from area station)			
Read request	12	0 0 0 1	0 0 1 0
Read cmd acknowledge	46	0 1 0 0	0 1 1 0
Read data request	62	0 1 1 0	0 0 1 0
Read data acknowledge	6A	0 1 1 0	1 0 1 0
Read end request	72	0 1 1 1	0 0 1 0
Read end acknowledge	3E	0 0 1 1	1 1 1 0
Read null acknowledge	42	0 1 0 0	0 0 1 0
Write commands (from System/7)			
Write	05	0 0 0 0	0 1 0 1
Write null	01	0 0 0 0	0 0 0 1
Write data	09	0 0 0 0	1 0 0 1
Write end	0D	0 0 0 0	1 1 0 1
Write responses (from area station)			
Write data request	51	0 1 0 1	0 0 0 1
Write data acknowledge	69	0 1 1 0	1 0 0 1
Write end request	71	0 1 1 1	0 0 0 1
Write end acknowledge	3D	0 0 1 1	1 1 0 1
Write null acknowledge	41	0 1 0 0	0 0 0 1
Write cmd acknowledge	45	0 1 0 0	0 1 0 1
Control commands (from System/7)			
Bypass	0B	0 0 0 0	1 0 1 1
Restore	03	0 0 0 0	0 0 1 1
Send sync	04	0 0 0 0	0 1 0 0
Begin diagnostic	08	0 0 0 0	1 0 0 0
End diagnostic	0C	0 0 0 0	1 1 0 0
2792 Reset	0F	0 0 0 0	1 1 1 1
Control responses (from area station)			
Bypass acknowledge	6B	0 1 1 0	1 0 1 1
Restore acknowledge	63	0 1 1 0	0 0 1 1
(response to send sync cmd)	47	0 1 0 0	0 1 1 1
Begin diagnostic acknowledge	68	0 1 1 0	1 0 0 0
End diagnostic acknowledge	4C	0 1 0 0	1 1 0 0

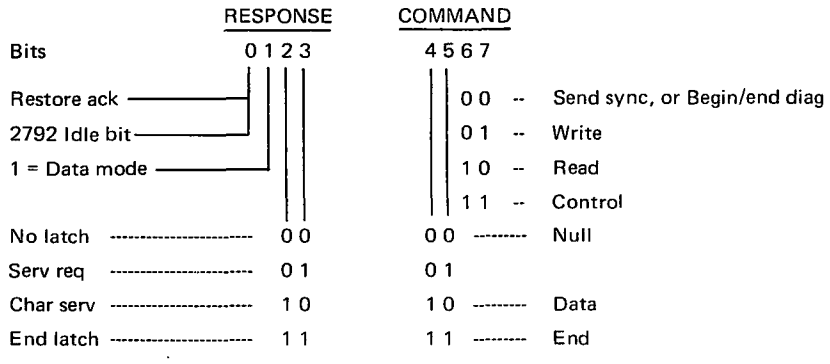


Figure 14-5. 2790 loop commands and responses

Read Commands

All read commands are generated by the program in the System/7 processor and sent to the area station by the 2790 Control.

Read: This code (hexadecimal 06) is sent after an area station has captured an any-address frame and the area station has returned a read request response. The code puts the area station adapter in data mode. The only valid response is read command acknowledge.

Read Null: This code (hexadecimal 02) specifies that the frame is available for any type of read request. This code should be generated by the program when no other command is to be transmitted. Four valid responses are: read null acknowledge, read request, read data request, and read end request.

Read Data: This code (hexadecimal 0A) is sent when a read data request response is received. The code causes the area station adapter to send the data byte again, so that a validity check can be performed, and also releases the area station adapter to prepare for the next character. The only valid response is read data acknowledge.

Read End: This code (hexadecimal 0E) is sent either as a result of receiving a read end request from the area station adapter or as a result of detecting an error that requires the operation to be terminated. This command removes the area station adapter from data mode, thus causing the operation in progress to be terminated in the area station adapter. The area station refers to the data byte to determine which indicator light to turn on. The only valid response to this command is read end acknowledge. When the command is sent to an area station local I/O adapter, the data byte contains an operator guidance character. (Refer to the manual *IBM 2790 Data Communication System Component Description,* Order No. GA27-3015, for a description of the operator guidance functions.)

Read Responses

All read responses are generated by the area station and sent back to the 2790 Control.

Read Request: This code (hexadecimal 12) is a response to a read null command when the area station captures an any-address frame to initiate a read operation. The code indicates that the area station wants to transfer information to the 2790 Control.

Read Command Acknowledge: This code (hexadecimal 46) is a response to a read command. The response indicates that the area station has received the read command and will start to read its data source (for example, a badge or a data entry unit).

Read Data Request: This code (hexadecimal 62) is a response to a read null command if the area station adapter has a data byte to send to the 2790 Control.

Read Data Acknowledge: This code (hexadecimal 6A) is a response to a read data command. The response indicates that the area station has executed the command. The area station response digit is the same as in read data request response. The controlling program should check the data byte received for validity against the one received earlier with the read data request.

Read End Request: This code (hexadecimal 72) is a response to a read null command if the area station adapter has determined either that the data transfer has come to a normal completion or that an error has been detected. The status of the operation is transmitted in the data byte. An all-zero status designates a normal end.

Read End Acknowledge: This code (hexadecimal 3E) is a response to a read end command. The response indicates that the area station adapter will terminate the operation according to the status in the data byte. The transaction is completed when the code is received by the 2790 Control. The program should check the data byte received (ending status) for validity against the one transmitted earlier with the read end command.

Read Null Acknowledge: This code (hexadecimal 42) is a response to a read null command if no action is presently required by the area station adapter and the area station is in data mode.

Write Commands

All write commands are generated by the program in the System/7 processor and sent to the area stations by the 2790 Control.

Write: This code (hexadecimal 05) initiates a write operation and puts the area station adapter in data mode. The only valid response is write command acknowledge. If the area station adapter cannot execute this command, it will make an end request when it receives the next command, a write null command. If a write command is sent to an area station local I/O adapter and the adapter is busy, the area station will return a data byte containing hexadecimal 80. The command should be sent repeatedly until a write command acknowledge response is received.

Write Null: This code (hexadecimal 01) specifies that the frame is available for any type of write request from the device that is addressed. There are three valid responses: write null acknowledge, write data request, and write end request.

Write Data: This code (hexadecimal 09) is sent to an area station when a write data request is received. This command is accompanied by a data byte, and the only valid response is write data acknowledge.

Write End: This code (hexadecimal 0D) is normally sent if a write data request response is received and there is no data left to be transmitted. The command can also be sent as a result of a program detected error, or as a result of a write end request response by the area station adapter after it detects an abnormal condition. The data byte with the write end command is used as a status field, with all 0's representing normal end. The only valid response is a write end acknowledge.

Write Responses

All write responses are generated by the area station and sent back to the 2790 Control.

Write Data Request: This code (hexadecimal 51) is a response to a write null command if the area station adapter is ready to receive a data byte.

Write Data Acknowledge: This code (hexadecimal 69) is a response to a write data command, and indicates that the area station has received the data byte. The program should check the data byte received for validity against the one transmitted earlier with the write data command.

Write End Request: This code (hexadecimal 71) is a response to a write null command if the area station adapter has detected an error or an unusual condition. The status is also sent in the data byte.

Write End Acknowledge: This code (hexadecimal 3D) is a response to a write end command. This response indicates that the area station adapter will terminate the operation according to the status in the data byte as modified by the area station adapter. An all-zero status represents normal end. The received status byte should be checked by the program for validity against the transmitted status byte.

Write Null Acknowledge: This code (hexadecimal 41) is a response to a write null command if no action is presently required by the area station adapter and the adapter is in the data mode.

Write Command Acknowledge: This code (hexadecimal 45) is a response to a write command. The response indicates that the area station adapter has received the command and that the adapter has inserted its status in the data byte. An all-zero status indicates that the operation will be executed. A nonzero status indicates busy, which can be returned only by the area station local I/O adapter.

Control Commands

All control commands are generated by the program in the System/7 processor and sent to the area stations by the 2790 Control.

Bypass: This code (hexadecimal 0B) can be used by the program as part of its diagnostic procedures. The command causes the area station to remove its shift register from the loop after the control byte has been received. This bypass operation eliminates the data byte from this frame, since there is a one-byte delay in the shift register. Since the area station removes the data byte from the frame, a short count results in the 2790 Control byte counter. However, the start byte of the next frame resets the byte counter.

While an area station is in a bypassed state, it can monitor the line, but it cannot insert information. The only valid response is bypass acknowledge.

Restore: This code (hexadecimal 03) is used by the program as part of its diagnostic procedures. The command causes the bypassed area station to insert its shift register in series with the loop after the control byte has been received. This insertion results in a repeat of the control byte (the restore command), which adds one byte to the active portion of the frame. A skip character (all 0's) is also substituted for the old data byte. The active frame returning to the 2790 Control contains six bytes instead of five. The fourth and fifth bytes contain the restore command and the sixth byte contains the skip character. The skip character causes the 2790 Control byte counter to skip a count.

If a restore command is sent to an area station that is not bypassed, the area station will return a restore acknowledge response, which enables the program to determine whether the command has been executed. The only valid response is the restore command itself immediately followed by a repetition of the restore command.

Send Sync: This code (hexadecimal 04) may be used by the program after an area station has been placed in diagnostic mode. When an area station in diagnostic mode receives a send sync command, the area station sends a continuous stream of sync characters (each containing hexadecimal 47) until it receives from the 2790 Control at least two characters in succession containing hexadecimal FF. This can be achieved by sending a frame with hexadecimal FF in all byte positions. The only response to this command is the fact that the area station generates a stream of sync characters.

The send sync command provides the ability to isolate a link (connecting two successive area stations) that is failing intermittently and causing the system's maximum error rate count to be exceeded.

The command allows predetermined data (sync characters) to be constantly transmitted by a selected area station without that area station transmitting what it is receiving.

2792 Reset. This code (hexadecimal 0F) is used to reset circuitry in the 2792 for initial startup and error reset conditions. It is recognizable only when used with an "all area station" address (hexadecimal 09) and is ignored by 2791/93 area stations.

Begin Diagnostic: This code (hexadecimal 08) can be used by the program as part of the area station diagnostic procedure. The addressed area station enters diagnostic mode when it receives this command. In diagnostic mode, the area station responds only to commands addressed to it. The only valid response is begin diagnostic acknowledge.

End Diagnostic: This code (hexadecimal 0C) is used by the program to end the area station diagnostic procedure. The area station is removed from diagnostic mode and returns to normal operation when this command is received. The only valid response is end diagnostic acknowledge.

Control Responses

All control responses are generated by the area station and sent back to the 2790 Control.

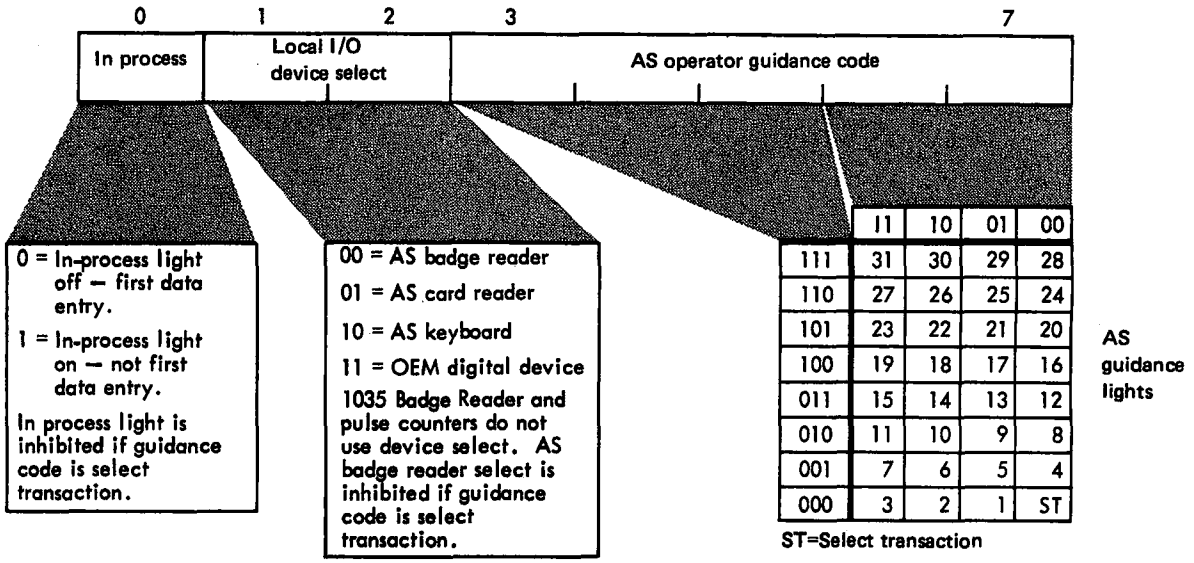
Bypass Acknowledge: This code (hexadecimal 6B) is a response to the bypass command and indicates that the area station has executed the command.

Restore Acknowledge: This code (hexadecimal 63) is a response to the restore command if the area station is already on line (not bypassed) when the command is received. This response is also inserted prior to the data byte if the area station is off line (bypassed) when the restore command is received.

Begin Diagnostic Acknowledge: This code (hexadecimal 68) is a response to the begin diagnostic command, and indicates that the area station has entered diagnostic mode.

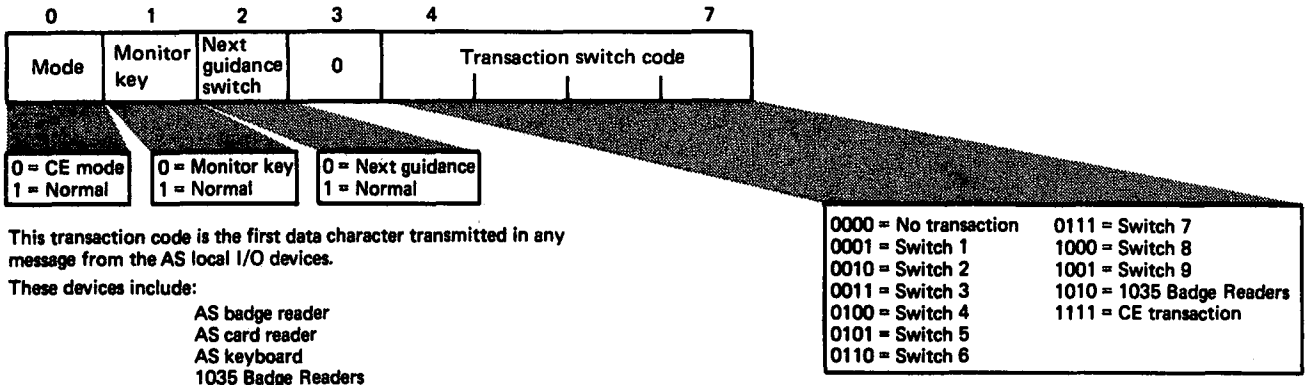
End Diagnostic Acknowledge: This code (hexadecimal 4C) is a response to the end diagnostic command, and indicates that the area station has returned to the normal mode of operation.

Area Station Local I/O Guidance Character



The guidance character is the second character transmitted in any message from the AS local I/O devices (except 1035 Badge Reader and pulse counters). In this case, the the guidance character reflects the "old guidance"--that is, the previous guidance character sent with a read end command to a local I/O device. A guidance character sent with a read end command to a local I/O device is considered to be "new guidance." It becomes "old guidance" when a new message is initiated.

Area Station Local I/O Transaction Code Byte

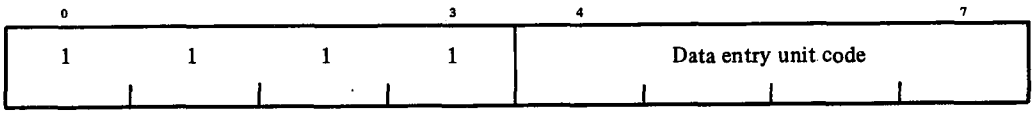


This transaction code is the first data character transmitted in any message from the AS local I/O devices.

These devices include:

AS badge reader
AS card reader
AS keyboard
1035 Badge Readers
AS transaction and next guidance switches
OEM digital device

Area Station Local I/O Guidance Character



Status character codes

Status character (hex)	*Associated with end request from area station means:*	*Associated with end command results in the following action at the area station:*
	Local I/O of area station	
00	Good end	Select transaction indicator
20	Diagnostic mode	No action
40	Release in data mode	Light repeat/clear
60	Release in diagnostic and data mode	Light repeat/clear
80	Invalid character	Light repeat/clear
A0	Invalid character and diagnostic mode	Light repeat/clear
C0	Overrun—card reader	Light repeat/clear
E0	Overrun in diagnostic mode	Light repeat/clear
All other	Accepted as guidance characters by local I/O device	

Status character (hex)	*Associated with end request from area station means:*	*Associated with end command results in the following action at the area station:*
	Data entry unit	
00	Good end	Release work station
20	Diagnostic mode	No action
40	Overrun	Error indication
60	Overrun in diagnostic mode	No action
80	EOB time-out	Error indication
A0	EOB time-out and diagnostic mode	No action
	Output adapter	
00	Good end	No action
20	Diagnostic mode	No action
80	Out of paper	No action
A0	Out of paper and diagnostic mode	No action

Critical Timing Conditions

Input/output adapters contained in 2791/2793 Area Stations are designed to operate at a data rate of 212 frames per second, spaced at equal intervals of 4.7 milliseconds. However, the time required for mechanical operations to take place within a 2790 system device may reduce the data rate from the maximum of 212 frames per second.

Figure 14-6 shows the 2790 devices that have critical timing conditions. Most of these critical timings apply only to a specific sequence of commands when they are addressed to the particular device. For example, a read data command frame addressed to a 1035 Badge Reader must be followed by two frames that are not addressed to the same 1035 Badge Reader before a read null command frame can be addressed to this badge reader. (A read null command frame can, however, be addressed to another device or another 1035 Badge Reader.)

Critical timing situations can be avoided by observing the designed data rates. These rates can be achieved by using every 45th frame to communicate with a 1053 Adapter, and every 9th frame to communicate any command sequence to other input/output adapters.

2791/2793 Adapter	Critical timing
Input adapter 1035 Badge Reader	Minimum of 2 frames* between a read data command frame and a read null frame.
Card reader	① Minimum of 2 frames* between a read data request frame and read data command frame. ② Maximum of 14 frames* between two consecutive read null frames (to avoid character overrun condition in 2790).
Data entry unit adapter	Maximum of 16 frames* between two consecutive read null frames (to avoid character overrun condition in 2790).
1053 Adapter	① Normal rate of 42 frames per second at equal intervals of 23.5 milliseconds. (Approximately 45 frames* between frames addressed to this adapter). ② Minimum of 133 frames* between two consecutive write null command frames that follow a write data command frame.

* Frames included in this count cannot address the adapter or device to which this timing applies.

Figure 14-6. 2790 critical timings

2792 Error Status

Status Characters (hex)	*Meaning*
00	Good status
01	Open loop
02	Data check
04	Control check
08	Parity error
10	Communications data error
20	Communications channel timeout
40	All area station command error

2790 Control Operation

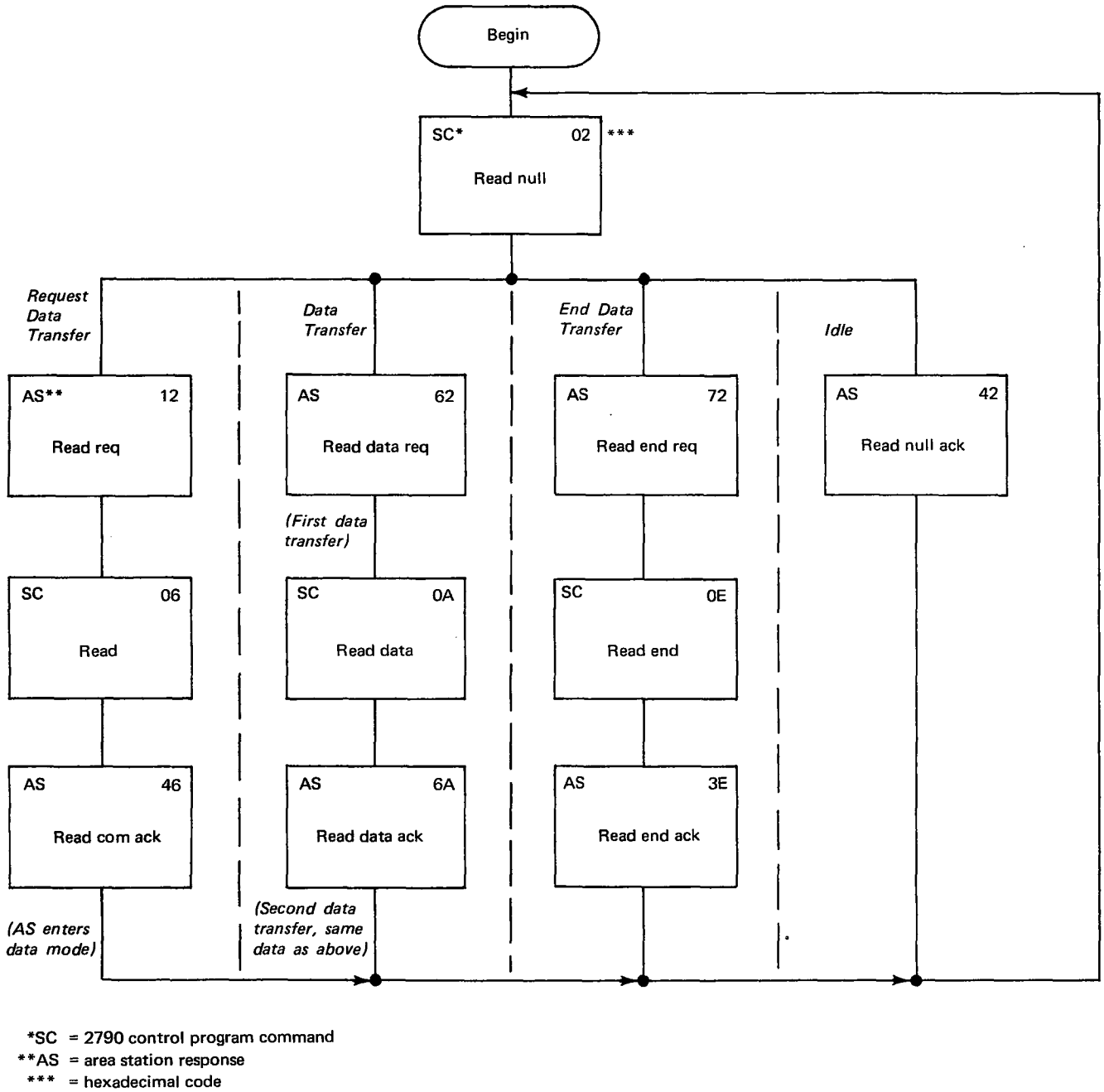
The 2790 Control uses two 4-byte buffers—an output buffer and an input buffer. The output buffer must be loaded with the area station address, device address, control, and data bytes to be transmitted on the 2790 loop. When the loop is not actively transmitting data (the four output buffer registers have not been filled by the program), a start byte followed by 29 sync bytes will be transmitted on the loop every 524 microseconds. When 30 bytes have been transmitted and the output buffer register has been filled, the next outgoing frame contains the start byte followed by the contents of the four-byte output buffer register. Twenty-five sync bytes follow this active frame, and then another start byte.

Frame cycles transmitted from the output buffer are received back in the 2790 Control in the four-byte input buffer. When the area station address, device address, control, and data bytes are received in the input buffer, the 2790 Control presents an interruption request to the System/7 processor. Errors are detected in the 2790 Control at the input buffer register.

Command Sequences

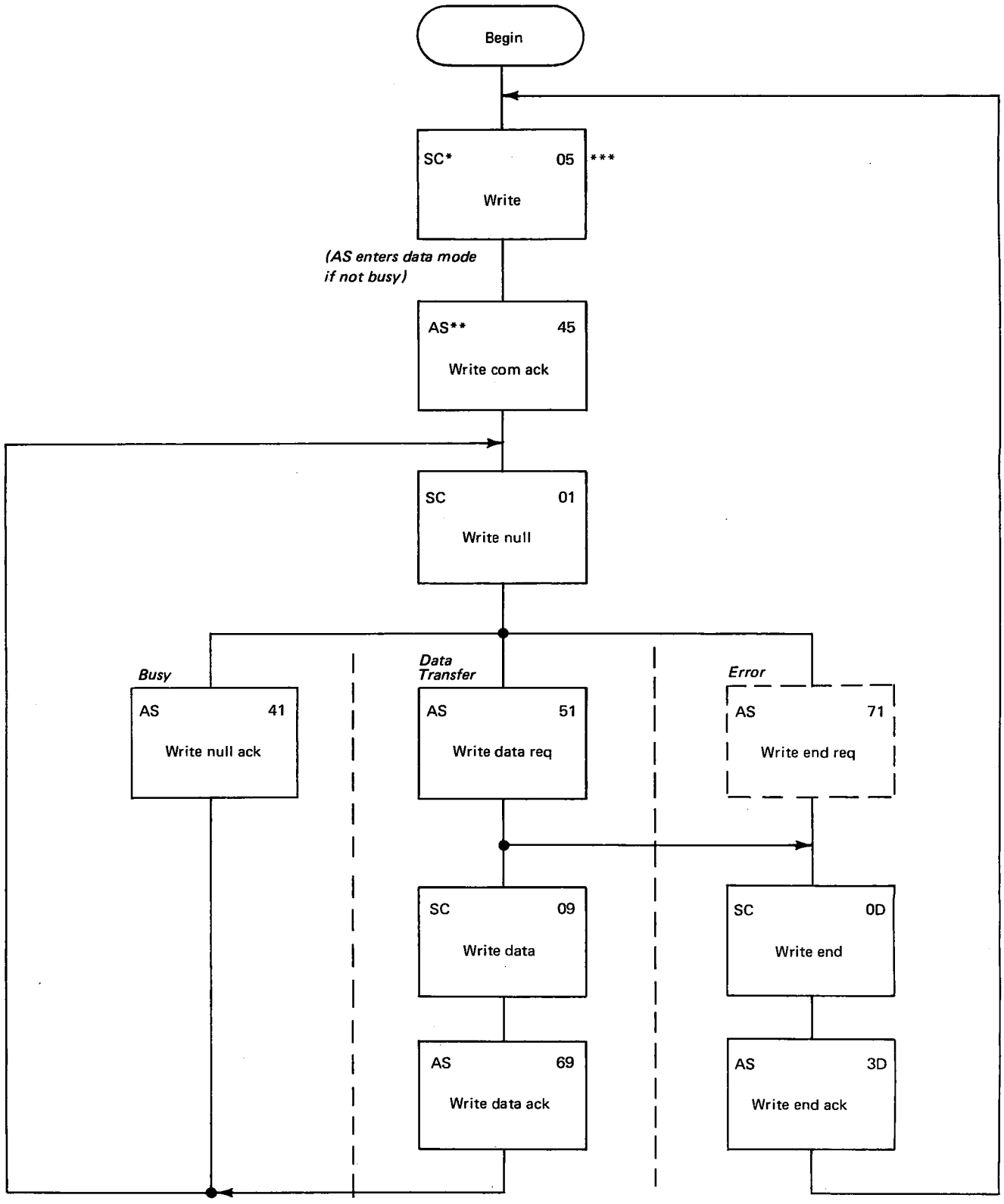
The 2790 Control must transmit several active frame cycles to perform a single operation. Figures 14-7 and 14-8 show read and write command sequences, respectively, and the proper area station responses to the commands. Critical timing conditions may exist for certain 2790 devices as shown in Figure 14-6.

For example, when no area stations require servicing, a frame containing an any-address code in the area station address byte and a read null command in the control byte is transmitted on the loop (see Figure 14-7).



*SC = 2790 control program command
**AS = area station response
*** = hexadecimal code

Figure 14-7. 2790 read sequences



*SC = 2790 control program command
**AS = area station response
*** = hexadecimal code

Figure 14-8. 2790 write sequences

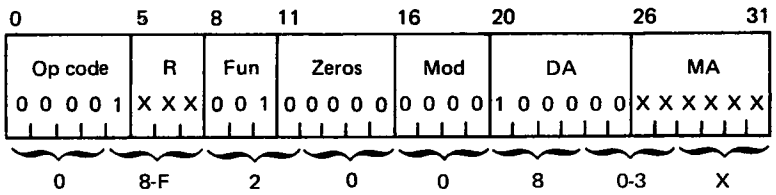
An area station requiring service looks for the any-address code and captures the frame by replacing the code with its discrete area station address. This not only identifies the area station requesting service, but also prevents any other area station from capturing this frame. The area station then inserts the address of the device that will be serviced into the device address field and inserts a read request response code in the control byte. The program then generates an outgoing frame containing the area station address, device address, a read command, and a 0 data byte. This transmission places the area station in data mode, which permits the requesting device to transmit data. The area station responds to the read command, and the program sends out a read null command. The area station responds to the read null command with a read data request when it has a data byte ready to be sent to the 2790 Control. This data byte accompanies the read data request response. The program then sends out a read data command, to which the area station responds with a read data acknowledge and, again, the data byte. This double transmission of the data byte allows for validation of every data byte transmitted by the area station.

Under normal operation the area station can initiate the read-end sequence when it has completed a data or transaction code entry. The sequence is started when the area station responds to a read null command addressed to it with a read end request and normal status in the data byte. The controlling program then sends out a read end command with either a guidance character or status information in the data field, depending on whether the addressed device is a data entry unit or a device on the area station local I/O adapter. Guidance characters apply only to the local I/O adapter device.

I/O Commands

The prepare I/O command is used as described in Chapter 4 under "Input/Output Instruction." Data transfers between the 2790 Control and the processor module are accomplished by immediate read and immediate write commands. Modifier field bits in the immediate read and write commands further define the operation to be performed. All I/O commands addressed to the 2790 Control function must have a device address of 100000. The module address varies, depending upon the physical location of the digital I/O module in the 5026 Enclosure. The 2790 Control has two 4-byte buffer registers, one for input frames and one for output frames. Two immediate write commands are used to fill the output buffer register with a frame for transmission, while two immediate read commands are used to read the frame in the input buffer register. The start byte (the first byte of a frame) is generated by the 2790 Control, so this byte is not loaded into the output buffer or read from the input buffer.

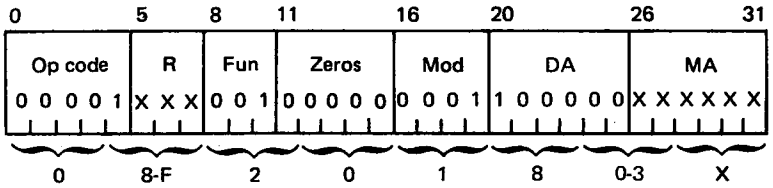
Write Address



This command stores the area station address and device address bytes into the first two bytes of the output buffer register in the 2790 Control. These two address bytes are transferred from the index register (R), or the accumulator if R=000.

The write-address command must be issued once to establish the desired addresses in the output buffer register. Any number of write-control-and-data commands, described next, can then be issued without having to issue another write address command. Another write address command must be issued only when the address(es) must be changed.

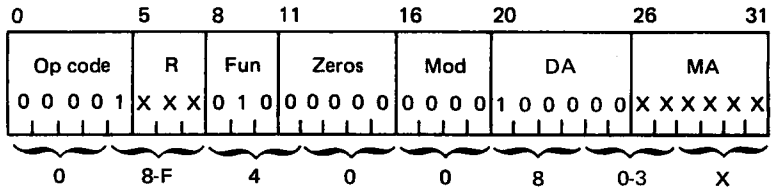
Write Control and Data



This command stores the control and data bytes into the last two bytes of the output buffer register in the 2790 Control. These two bytes are transferred from the index register (R), or the accumulator if R=000.

After these bytes are loaded into the output buffer register, the 2790 Control transmits to the 2790 system an active frame cycle that contains the information in the output buffer.

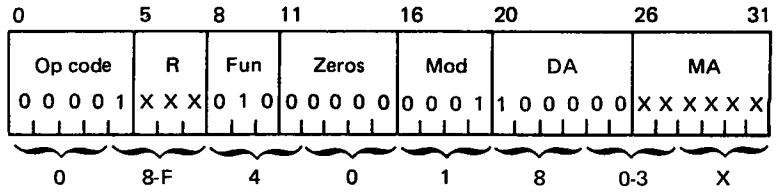
Read Address



This command stores the first two bytes of the input buffer register (area station address and device address) into the index register (R), or the accumulator if R=000.

These addresses are obtained by the program so that the contents of the incoming frame can be compared with the contents of the outgoing frame for validity and for future use in outgoing frames. The addresses may also be placed by the program in the next active outgoing frame that is generated.

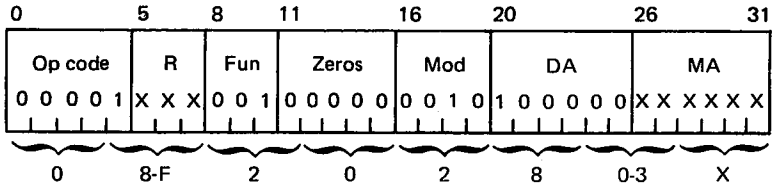
Read Control and Data



This command stores the last two bytes of the input buffer register (control and data bytes) into the index register (R), or the accumulator if R=000.

The control and data bytes should be checked for valid responses and data, respectively, every time a frame is received from the area station determined by a read address command (described previously). The results of this check may also determine the control and/or data that should be placed in the next outgoing frame. (See Figure 14-5 for valid 2790 Control commands and responses.)

Set Diagnostic Mode

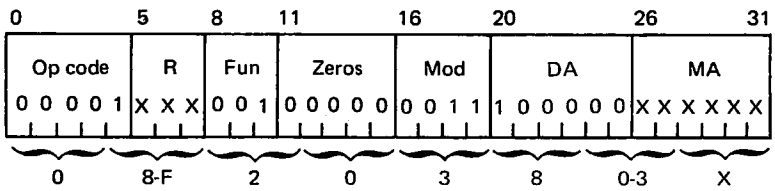


This command places the 2790 Control function in one of four states, depending upon the value of bits 14 and 15 in the index register (R), or the accumulator if R=000.

The four possible states set by this command are:

Bits		*State*
14	*15*	
0	1	Diagnostic mode. The 2790 Control is set in diagnostic mode of operation for use with the set-input-buffer command described later. ISW bit 5 is also set on to indicate that the diagnostic mode is effective. While in the diagnostic mode, the 2790 Control constantly checks for the presence of sync characters in bytes 5 to 29 of a frame cycle. Any missing sync bytes are indicated by setting on bit 6 in the ISW.
1	0	Loop bypass. The 2790 loop is bypassed. Output from the output buffer register is connected directly to the input of the input buffer register.
1	1	Diagnostic and loop bypass. Output frames from the 2790 Control enter directly into the input buffer register and are checked for sync characters in bytes 5 to 29.
0	0	Normal operation. This code resets the above three states and removes the 2790 Control from the diagnostic and/or bypass modes of operation.

Set Input Buffer Control



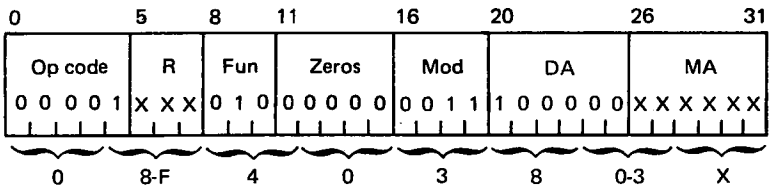
This command loads a sync character (hexadecimal 47) or a character containing all 1's into the input buffer register, depending on the setting of bit 15 in the index register (R), or the accumulator if R=000. A set-diagnostic-mode command must set the 2790 Control function in diagnostic mode before executing a set-input-buffer-control command.

Bit 15 in the R register can be set as follows:

Bit 15 = 0	An interruption request is presented if an inactive frame cycle is received (bytes 2 to 5 are sync characters).
Bit 15 = 1	The input shift register shifts logical 1's into the input buffer register. When all positions in the register contain 1's, an interruption request is presented to the processor module. Since the 2790 Control is in diagnostic mode, this "forced" shifting checks the input circuits without using the output portion of the function.

A read-address or read-control-and-data command would normally follow the set-input-buffer-control command when the interrupt is requested by the 2790 Control.

Read ISW



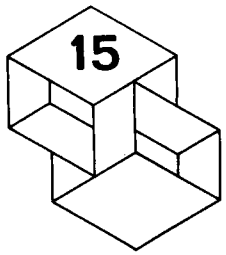
This command stores the 16-bit interrupt status word (ISW) into the index register (R), or the accumulator if R=000.

Interrupt Status Word (ISW): The 16 bits in the interrupt status word are set to indicate operating status of the 2790 Control function and to detect errors that occur after a write-control-and-data command is completed, but while the device is still busy. The ISW bits are reset on the first selection of the 2790 Control function after the interruption request has been accepted by the processor module.

Significant bits in the 2790 Control ISW and their meanings are:

Significant Bits	*Meaning*
3	Frame parity error. One of the four bytes of an active frame received in the input buffer had a parity error.
4	No input detected. The 2790 Control input circuits did not receive an input pulse from the loop within 1.024 ms.
5	Diagnostic mode. The 2790 Control is in the diagnostic mode of operation. This is not an error.
6	No sync bytes. A byte other than a sync byte was detected in bytes 5 to 29 of a frame cycle. This is valid only when the 2790 Control is in diagnostic mode, and is not an error.
7	Bypass mode. The 2790 Control function is in the bypass mode of operation. This is not an error.
12	Device busy. The 2790 Control is transmitting an active frame and is waiting for the interrupt from the frame returning in the input buffer.
13	Device end. The 2790 Control has completed the requested operation.

CHAPTER CONTENTS



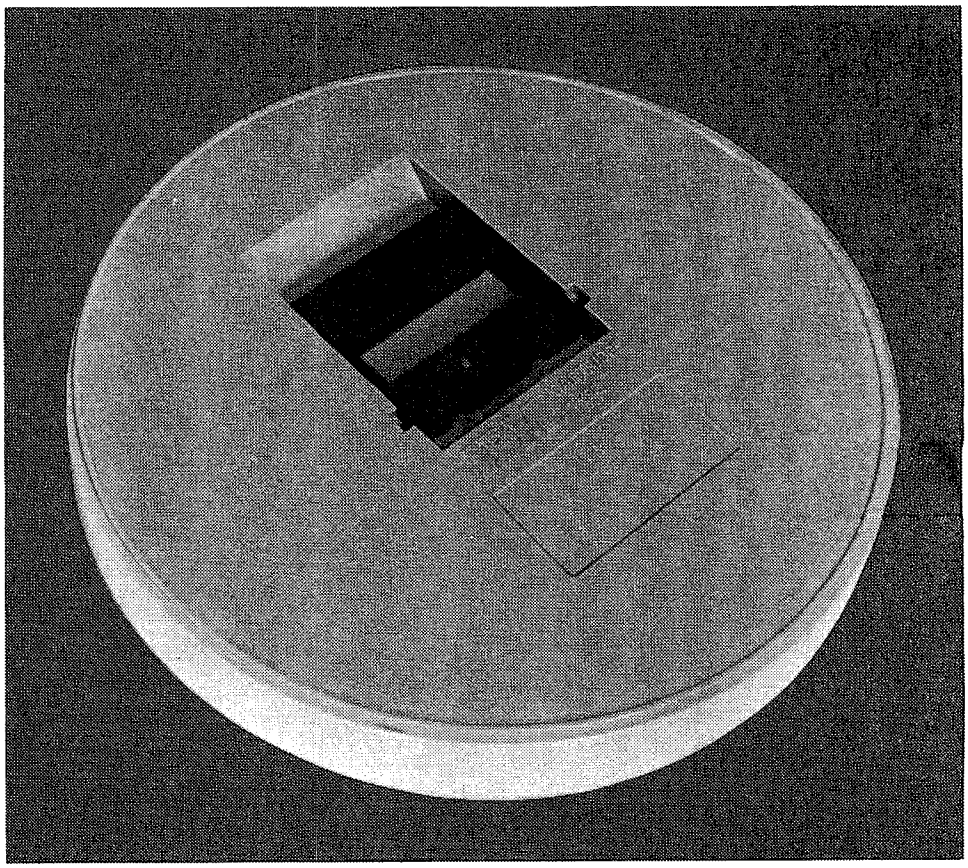
FUNCTIONAL DESCRIPTION

The IBM 5022 Disk Storage Module is a bulk data storage device that provides the System/7 with direct-access storage capability. Models 1 and 2 of the disk storage module can store a maximum of 2,457,600 data words; models 3 and 4 can store a maximum of 1,228,800 data words.

Models 1 and 2 of the disk storage module incorporate two disks, one above the other. The upper disk is contained in a disk cartridge (Figure 15-1) that protects the disk when it is removed from the drive and used for offline storage. Models 3 and 4 incorporate a single disk that cannot be removed.

As many as 12,288 words (model 1 or model 2) can be transferred to or from the disk without repositioning the read/write elements of the module. Data is transferred to and from the disk at a maximum rate of 99,500 words per second. Models 1 and 3 differ from models 2 and 4, respectively, only in the time required to move the access mechanism. This time difference is described later in this chapter under "Seek Time."

Direct program control (DPC) for data operations is standard equipment on all four models. An optional feature, 5022 Disk Cycle Steal (CS), is available. CS allows transfer of up to 3,072 words with only one read or write command. DPC is mutually exclusive with CS.



BR0433

Figure 15-1. Disk cartridge

Operator Controls

The following operator controls are located on the front of the disk storage module.

FILE WRITE Switch

The setting of this switch determines if data can be written on the disk in the module. When the switch is in the normal position, both reading and writing can be performed. When the switch is in the write inhibit position, only read commands are accepted by the disk storage module.

IPL Switch

This switch is operable only on models 1 and 2 of the disk storage module. It selects the upper or lower disk in the module that will be used in an initial program load (IPL) operation for the System/7.

DRIVE Switch

This switch turns power on or off to the disk drive in the module if system power is on. The drive must be turned off in order to change disk cartridges.

FILE READY Indicator

This indicator turns on when the disk file can be used by the System/7 program. It remains on until the disk file is turned off or a file not ready condition occurs. A file not ready condition is caused by errors detected in the disk storage module. For details of these error conditions, see "Error and Status Conditions" in this chapter.

Data Security

Data stored on the 5022 Disk Storage Module can be given additional security by use of the file write switch located on the front of the module. Data on the disk(s) cannot be changed by a program if the file write switch is set to prevent writing on the file, as described previously under "Operator Controls."

Each surface of each disk contains 204 tracks. The tracks that are related to each other in the vertical plane on a single disk are considered to form a cylinder, as shown in Figure 15-2. The corresponding cylinders on both disks have the same cylinder number.

Cylinders are numbered 0 through 203. Cylinder 203 is reserved for customer engineer use. Cylinders 1 through 3 are used as alternate cylinders if a track in cylinders 4 through 202 is found defective. Cylinder 0 and cylinders 4 through 202 are the normal data cylinders.

Sectors within a track are identified by their physical position on the track with relation to the index mark and by the surface of the disk on which they reside.

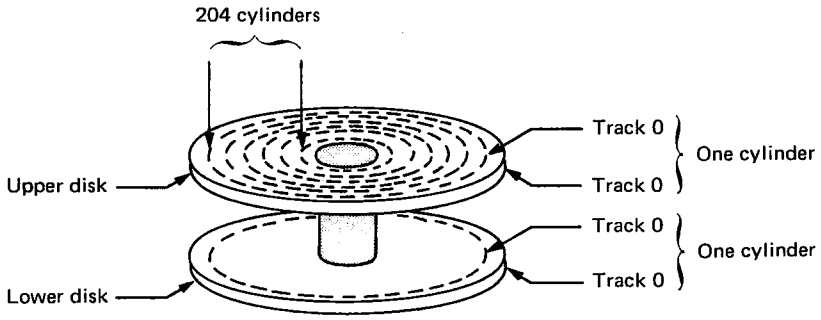


Figure 15-2. Cylinder concept

Track Format

Each track is divided into 24 sectors, as shown in Figure 15-3. Each sector in a track has its own individual address and is made up of an address mark, sector identifier, data field, and some gaps.

Index Mark: A mark that is fixed for each disk and provides orientation information to the module circuits. It is the starting point for every track.

AM (address mark): A specially written group of bits used to indicate the start of a new sector.

ID (sector identifier): A three-word identification that is unique for each sector on the disk. The format of this identification is described later in this chapter under "Sector Identifier Format."

Data: An area of the sector that contains 128 words of data plus three check characters.

Gaps: Specially written areas on the disk that are used to separate and define the other elements of the sector.

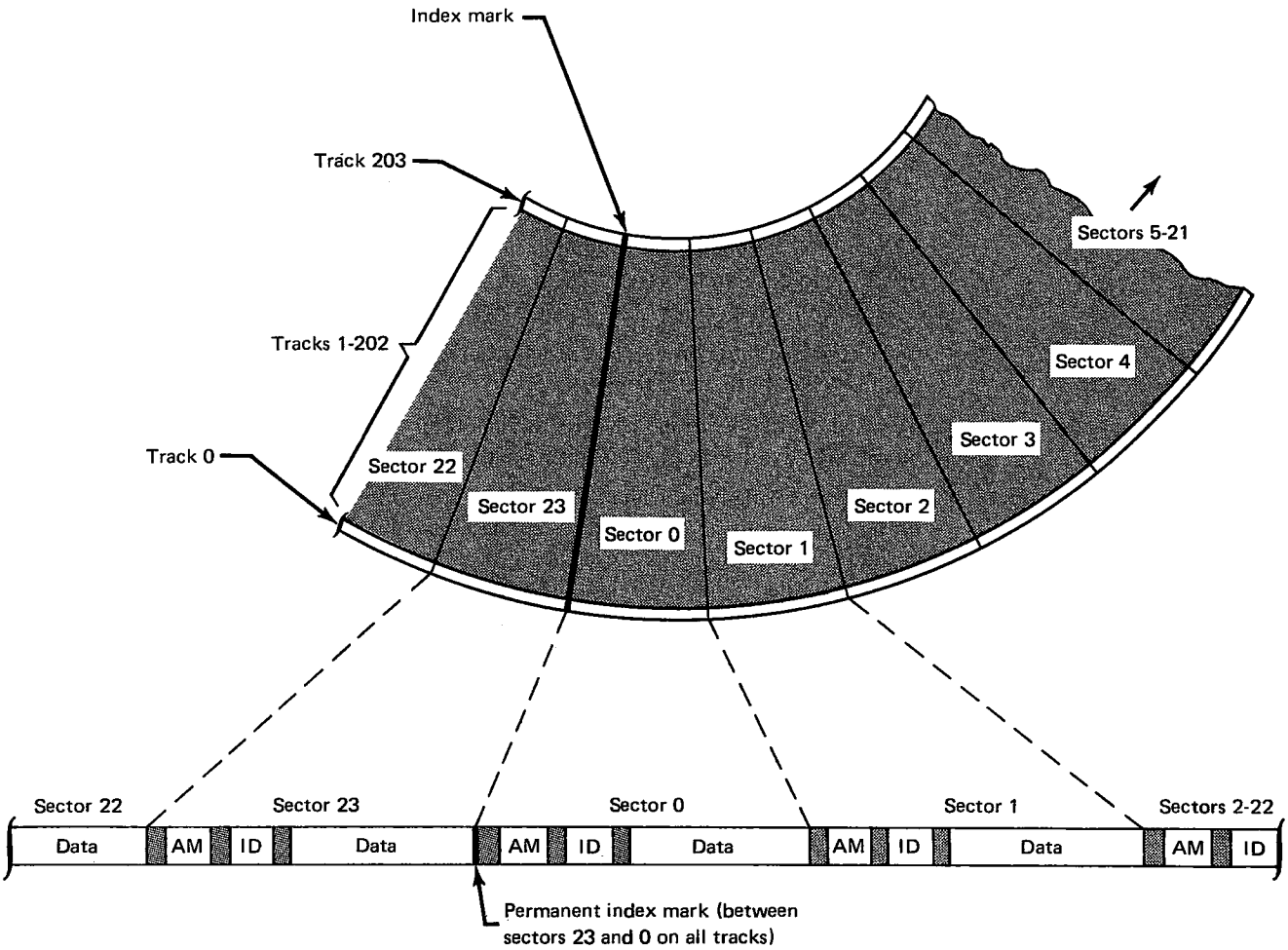
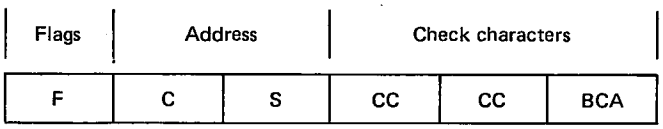


Figure 15-3. Track and sector format

Sector Identifier Format

Each identification area recorded on a sector contains six 8-bit bytes—three bytes of flags and address information and three bytes of check information:



F (flag byte): Contains flagging information in bits 6 and 7 to indicate the condition of the track surface. All other bits in this byte should be 0. Flagging information is covered in more detail under "Flagging," later in this chapter.

C (cylinder byte): Contains the binary number corresponding to the physical location of the track on the disk.

S (sector byte): Contains the number of the sector. Sectors on the top surface of the disk have sector numbers from 0 through 23. Sectors on the lower surface of the disk have sector numbers from 32 through 55.

CC (cyclic check): Contains an automatically generated check character used by the file circuits to verify the data field.

BCA (bit count appendage): Contains an automatically generated check character used by the file circuits to verify the entire read or write operation.

The address of any individual sector is contained in the first three bytes (F, C, and S) of the sector identifier. This address applies only to the disk on which these bytes are located. Sectors occupying the same physical location on all disks have identical binary numbers in the cylinder and sector bytes. The sectors on the upper surface of the disk are numbered 0 through 23, starting with the index mark, and the sectors on the lower surface are numbered 32 through 55. A specific sector, then, is addressed by specifying the upper or lower disk, a cylinder number, and the sector number.

OPERATING AND PROGRAMMING RESTRICTIONS

The disk storage module operates only when (1) the drawer containing the disk file drive is closed, and (2) a disk cartridge is properly installed on the drive (models 1 and 2 only). The drawer cannot be opened unless System/7 power is on and the file ready indicator is turned off. The drawer should be kept closed at all times unless a disk cartridge is being inserted or removed. A disk cartridge must be stored in the operating environment for at least two hours before the cartridge is used for processing.

On a disk storage module without CS, a data overrun condition occurs if an interruption request for data (called a data-service interruption request) is not serviced within 40 microseconds. Although the disk storage module can be prepared to request interruptions on any priority level, care should be exercised in assigning a priority interruption level to the disk. It is advisable to assign the disk storage module to the highest priority level in order to avoid a data overrun condition. (If a data overrun condition occurs, data is not lost, but the data transfer rate is reduced due to the rotational delay of the disk before the I/O operation can be started again.)

On a disk storage module with CS, an overrun occurs when the channel fails to respond in time to a request for service from the module. In a typical operating environment, occurrence of an overrun should be extremely rare. It can occur only when other modules cause a high level of activity on the channel during a read or write operation on the disk storage module. An overrun condition results in terminating the operation and presenting an interruption. Overrun (bit 2), any error (bit 3), channel end (bit 11), and device end (bit 13) will be present in the ISW.

The recommended recovery procedure is as follows:

1. Read the residual address. The command to do this is an immediate read command with a modifier of hex F (1111). The command may be used by the program to determine the contents of the data address register which is held and updated by the disk storage module subchannel. If an overrun occurs, the data address register will contain the address of the word where the data service failed to occur. (See "Read Residual Address" command in this chapter.)
2. Determine the sector where the overrun occurred.
3. Compute the count, data address, and sector address.
4. Reinitialize the device control block.
5. Issue the read/write cycle steal command.

DPC FILE OPERATIONS

I/O operations for the disk storage module are performed by read, write, and seek commands. Each I/O command specifies a register that contains data or control information. (The contents of the specified register are described later under the I/O command descriptions.) Error and status conditions of the disk storage module are indicated by status words that can be read at any time. (See "Error and Status Conditions," later in this chapter.)

Read Data

Two separate commands must be used to read data from a selected disk. First, a read initialize command is used to find the sector from which data is to be read. When that sector is found, the disk storage module begins reading the data from the sector into a four-word buffer in the disk storage module. When the buffer is full, the disk storage module presents a data-service interruption request to the processor module. At this time, only read data commands are required to complete the read data operation.

Four read data commands must transfer the buffer contents to processor storage within 40 microseconds. The disk storage module again loads the buffer and interrupts the processor, and the program again transfers the buffer contents to processor storage. This sequence continues until 128 words have been read from the disk sector and their check character is verified. (The disk storage module automatically verifies the check characters used in reading the sector ID and the data field from the disk.)

If fewer than 128 words are to be transferred to processor storage, a terminate command must be used instead of a read data command. The terminate command prevents the disk storage module from requesting any more data-service interruption requests. However, the disk storage module continues to read data from the disk sector in order to verify the check character for the entire 128-word data field.

Write Data

Two separate commands must be used to write data on a selected disk. First, a write initialize command is used to find the desired sector on which data is to be written. When that sector is found, the disk storage module presents a data-service interruption request to the processor module. At this time, only write data commands are required to complete the write data operation.

Four write data commands must transfer data to fill the four-word buffer in the disk storage module within 40 microseconds so that the module circuits can write the data on the disk. When the four words are written on the disk, the disk storage module again interrupts the processor and the program again fills the four-word buffer. The disk storage module continues this sequence until 128 words have been written on the disk sector. Then, the disk storage module circuits write the check character that was automatically generated during the writing of the 128 data words.

If fewer than 128 words are to be transferred to the disk storage module, a terminate command must be used instead of a write data command. The terminate command prevents the disk storage module from requesting any more data-service interruption requests. Upon receipt of the terminate command, the disk storage module writes data words containing 0's in the remainder of the sector, followed by the corresponding check character for the entire 128-word data field.

CYCLE STEAL FILE OPERATIONS

I/O operations for the disk storage module with cycle steal (CS) are performed by read cycle steal, write cycle steal and seek commands. The read/write commands make invalid the following DPC commands:

Read initialize
Read data
Write initialize
Write data
Terminate

Issuing those DPC commands to a CS disk storage module causes bit 1 (command reject) to be set in the DSW. The other DPC disk commands are also used by the CS feature.

One CS I/O command can read or write up to 24 sectors (3,072 words) of information. The I/O command specifies a register that contains the starting address of the device control block (DCB). The DCB contains the information needed to perform a read cycle steal or write cycle steal operation. (The contents of the DCB are described later in this chapter under "I/O Commands.")

Issuing read cycle steal or write cycle steal commands to a processor without the CS feature causes a program check.

Engineering Note. The addition of the cycle steal feature to the 5022 Disk Storage Module reduces the processor loading factor from 33% to approximately 4% (percentages represent processor loading during data transfer operations with the disk storage module located in an A2, C3 or C6 enclosure). In addition, the cycle steal feature reduces the load on the channel and on the I/O interface; therefore, system throughput is increased.

Figure 15-4 illustrates the reduction in system loading during full-sector 5022 data operations.

Mode of operation	Percent of system bandwidth*			
	A2/C3/C6 Enclosure		D3/D6 Enclosure**	
	Processor	Channel	Processor	Channel
DPC	33%	24%	63%	46%
CS (write cycle steal)	4%	17%	4%	30%
CS (read cycle steal)	4%	17%	4%	45%

*Percent of bandwidth consumed during full-sector data transfers. Percentages are approximate and vary depending on the location of the disk storage module within the enclosure and the user's coding.

**Enclosure located at its maximum distance of 200 ft.

Figure 15-4. System loading–full-sector operations

Seek

A seek operation moves the read/write heads across the disk surface so that the contents of all 204 cylinders can be accessed. Any number of cylinders can be crossed with one seek command. A seek command also determines which disk (upper or lower) and which read/write head (upper or lower) will be used by subsequent read or write commands. Therefore, in some cases, the seek command does not move the access mechanism, but merely selects the disk and/or the read/write head to be used by a subsequent command.

Seek Time

Figure 15-5 shows the minimum, maximum, and average time required to access data for all models of the disk storage module. Seek time for models 1 and 3 can also be determined from the following formulas:

Seek time for one track = 32 ms
Seek time for two or more tracks = 36 ms + 3.42 (N–2) ms

where N = number of tracks to be crossed.

Seek time for models 2 and 4 cannot be calculated from a convenient formula because the times are not necessarily the same for both forward and reverse operations. The more important seek times for models 2 and 4 are shown in Figure 15-5.

Disk module	Cylinder seek time		Average data access *
	Minimum	Maximum	
Model 1	32 ms	710 ms	275 ms
Model 2	28 ms	255 ms	146 ms
Model 3	32 ms	710 ms	275 ms
Model 4	28 ms	255 ms	146 ms

* Average access times are based on an access of 66 cylinders and include 20 milliseconds average rotational delay.

Figure 15-5. Cylinder seek times

INITIAL PROGRAM LOAD (IPL)

An initial program load (IPL) operation can be performed from a disk storage module to System/7 storage. The IPL can be started either by pressing the program load key on the operator console or by the auto restart function.

Prior to starting the IPL operation, certain operator switches must be set accordingly.

- *System/7 operator console.*
  The load unit address switches must be set to the module address containing the disk to be used. The host attach switch must be set to either the disable or the enable position.
- *Addressed disk storage module.*
  On model 1 or model 2, the IPL switch must be set to select the disk (upper or lower) to be used in the module.

When the IPL operation starts, the System/7 processor is reset, the load light on the operator console turns on, and the addressed disk storage module moves the read/write heads to cylinder 0 to locate sector 0 on the selected disk. The contents of sector 0 are then read from the disk and transferred into processor storage locations 0 to 127. After 128 words have been transferred, the processor begins executing instructions at location 0 on priority level 3. When execution begins, the level 3 accumulator contains the IPL indicator in bit 0 and the module address of the disk storage module in bits 10 to 15. Bit 0 will be set off if the IPL was started by pressing the program load key; it will be set on if the IPL was started by the auto restart function.

If an error is detected during the IPL operation just described, the processor stops and the load light remains on. Operator attention is required to restart the IPL operation.

I/O COMMANDS

The 5022 Disk Storage Module is programmed using the prepare I/O and halt I/O commands as described in Chapter 4 under "Input/Output Instruction." Read and write commands transfer data and control information between the disk storage module and the processor module. The setting of the modifier field bits in the command further defines the operation to be performed.

In all I/O commands addressed to the disk storage module, the device address field is not used, but should be set to 0. The module address varies depending upon the physical location of the I/O module in the 5026 Enclosure. Disk I/O execution time also varies depending on the location of the disk storage module within the 5026 Enclosure. Timing relationships for the possible locations of the disk storage module are indicated by t (x) in Figure 15-6. The time value (t) increases from minimum to maximum within the appropriate range of PIO time given in Figure 15-6. For estimating purposes, a straight line interpolation can be used to approximate the value of t for any location.

Note that, although 16-bit control information words are transferred between processor storage and the disk storage module, the disk file circuits correlate the identifier fields in the 16-bit word with the corresponding fields in the proper ID bytes on the disk. This correlation is covered in more detail under "Flagging," later in this chapter.

Enclosure	*Range of PIO time (nanoseconds)*
5026 A2	1900–2300
5026 C3	2000–2400
5026 C6	2000–2500
5026 D3 or D6*	2500–3100
5026 D3 or D6**	3300–3900

*shortest external cable length
**longest external cable length

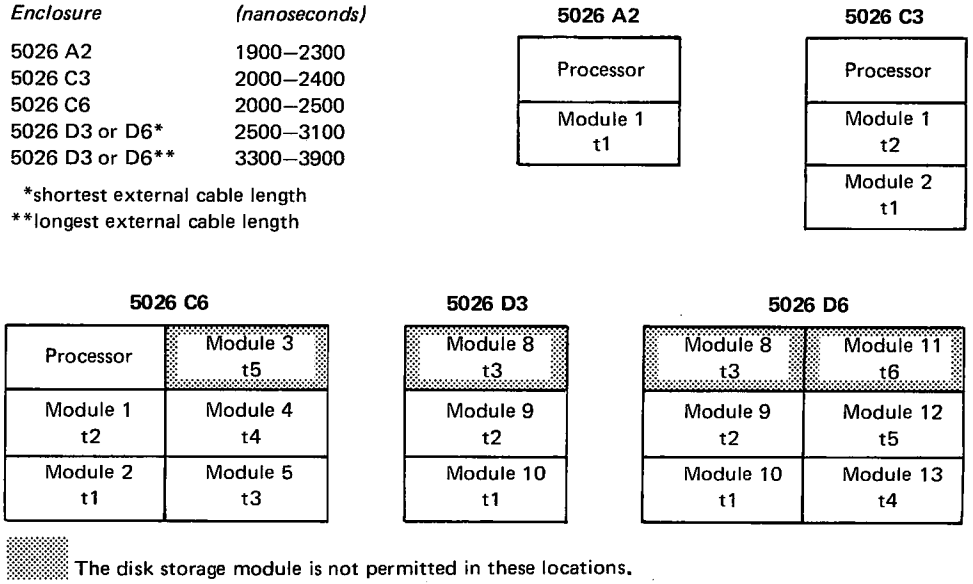
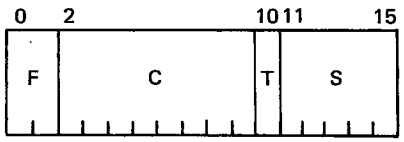


Figure 15-6. Disk storage module I/O execution time

Control information contained in the R-register is used to identify a sector and has the following format for the read initialize, read-sector-ID, read verify, and write initialize commands.



The F-bits (bits 0 and 1) are flag bits that indicate the condition of the disk track. Bit 0=1 indicates a defective track; bit 1=1 indicates an alternate track.

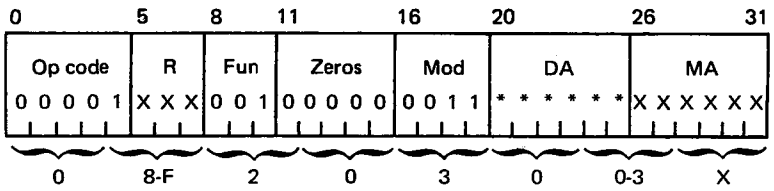
The C-bits (bits 2 through 9) contain the cylinder number.

The T-bit (bit 10) is an extension of the sector number, indicating a sector located in either an upper or lower track. T=0 indicates an upper track; T=1 indicates a lower track.

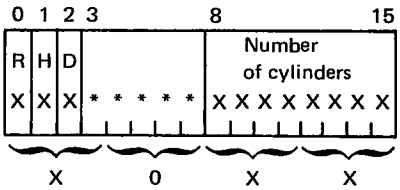
The S-bits (bits 11 to 15) contain the sector number to be used in the specified track.

Note: The read/write head (upper or lower) to be used must be selected by a prior seek command or load-next-sector-ID command.

Seek



The read/write heads of the disk drive move the number of cylinders and in the direction specified by the control information word contained in the index register (R), or the accumulator if R=000. The control information word has the following format:



For models 1 and 2 of the disk storage module, the R-bit selects the disk (upper or lower) for use by a subsequent read or write command addressed to the module. R=0 selects the upper disk; R=1 selects the lower disk. For models 3 and 4, the R-bit always equals 1.

The head bit (H) selects the read/write head to be subsequently used on the selected disk. H=0 selects the upper head; H=1 selects the lower head.

The direction bit (D) specifies the direction in which the heads are to be moved. D=1 specifies the forward direction (toward the center); D=0 specifies the reverse direction (away from the center).

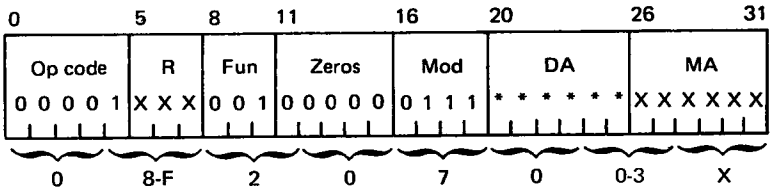
Bits 8 to 15 specify the number of cylinders to be crossed.

Upon completion of the seek operation, or if a seek error condition occurs, a device-end interruption request is presented to the processor module.

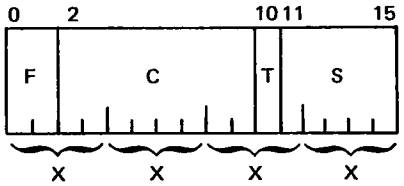
Note: The disk and/or read/write head are selected for a subsequent read or write command only through the seek command or the load-next-sector-ID command.

Recalibrate Operation: A recalibrate operation returns the access mechanism to cylinder 0 and selects a specified disk and head in the disk storage module. The recalibration is performed by a seek command that specifies moving the access mechanism in the reverse direction across 224 or more cylinders (bits 8 to 15 of the control information word equal 1110 0000 or greater).

Write Initialize (DPC)



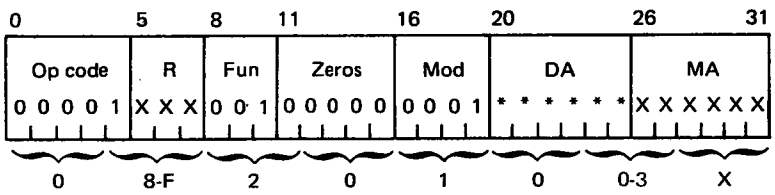
This command places the disk storage module in the write data mode of operation and searches for the sector identifier specified by the control information word contained in the index register (R), or the accumulator if R=000. The control information word has the following format:



Sector identifiers are read on the track previously specified and compared to the identifier specified by the control information word. When the specified sector is found, a data-service interruption request is presented to the processor module. If the sector is not found after at least one complete revolution of the selected disk, the no-record-found bit (bit 5) is set in the interrupt status word and an interruption request is presented to the processor.

Upon completion of an error-free write initialize command, the disk storage module can accept only a write data or terminate command. Write data mode is reset by either an end-of-sector or an any-error interruption request.

Write Data (DPC)

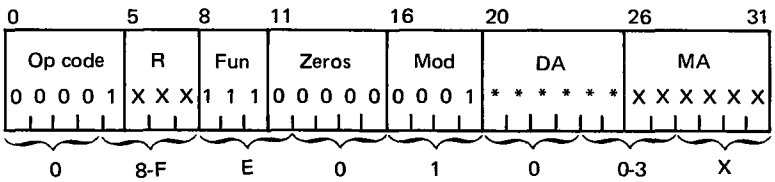


This command transfers one 16-bit data word from the index register (R), or accumulator if R=000, to the disk file data buffer. The write data command is valid only after a write initialize command has placed the disk storage module in the write data mode of operation.

Data-service interruption requests occur at 40-microsecond intervals. Four write data commands must fill the disk file data buffer within this interval to avoid a data overrun condition. If the data buffer is not filled within 40 microseconds, the overrun error bit (bit 2) is set in the interrupt status word and a device-end interruption is presented. Once the data buffer is filled, the disk storage module is busy to any more write data commands until the next data-service interruption request occurs.

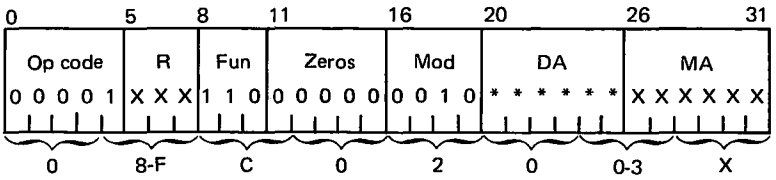
After 128 data words and the appropriate check character for the entire data field are successfully written, the end-of-sector bit (bit 7) is set in the interrupt status word and an interruption request is presented. (See "Read Data" for programming note.)

Write Cycle Steal (Cycle Steal Feature)



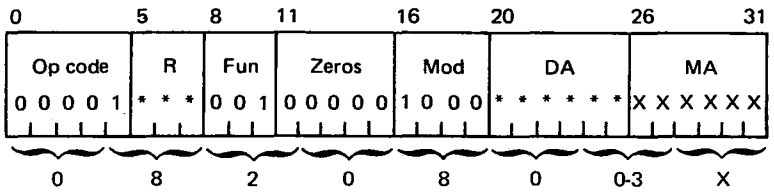
This command causes the device to cycle steal data from storage and write it on the disk. The index register (R), or the accumulator if R=000, contains the starting address of the DCB. The DCB contains the information needed to perform the operation. The disk module is busy to all commands, except status reads, from the time the command is issued until the channel-end interruption.

Read Verify Cycle Steal (Cycle Steal Feature)



This command is used to validate data written on the disk. No data is transferred to storage during its execution. The command would normally use the same DCB as a previous write cycle steal command. The index register (R), or the accumulator if R=000, contains the starting address of the DCB. The DCB contains the information needed to perform the operation. The disk module is busy to all commands, except status reads, from the time the command is issued until the channel-end interruption.

Terminate (DPC)



This command suppresses data-service interruption requests. The terminate command is valid only when the disk storage module is in the write data or read data mode of operation.

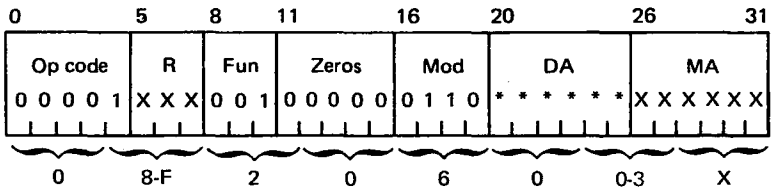
When the disk storage module is in write data mode, 0's are inserted into the remainder of the data field on the selected sector and the appropriate check character for the entire data field is written at the end of the data field.

When the disk storage module is in read data mode, the remaining data field is read from the selected sector and the check character is verified, but no additional data transfers to the processor module take place.

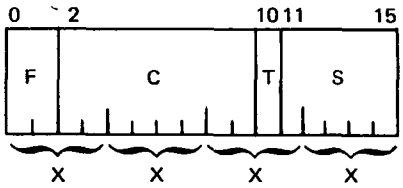
The end-of-sector bit (bit 7) is set in the interrupt status word and an interruption request is presented after the check character is either read or written.

Note: The terminate command may specify an index register (R), or the accumulator if R=000. In write data mode, the contents of the register specified by R are ignored by the disk storage module during execution of the terminate command. In read data mode, the contents of the register specified by R are not changed. (See "Read Data" for programming note.)

Read Initialize (DPC)



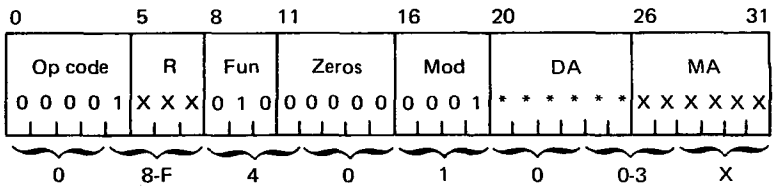
This command places the disk storage module in the read data mode of operation and searches for the sector identifier specified by the control information word contained in the index register (R), or the accumulator if R=000. The control information word has the following format:



Sector identifiers are read on the track previously specified and compared to the identifier specified by the control information word. When the specified sector is found, the first four data words in the sector are read from the disk into the file data buffer and a data-service interruption request is presented to the processor. If the sector is not found after at least one complete revolution of the selected disk, the no-record-found bit (bit 5) is set in the interrupt status word and an interruption request is presented to the processor.

Upon completion of a successful read initialize command, the disk storage module can accept only a read data or terminate command. Read data mode is reset by either an end-of-sector or an any-error interruption request.

Read Data (DPC)



This command transfers one 16-bit data word from the disk file data buffer to the index register (R), or the accumulator if R=000. The read data command is valid only after a read initialize command has placed the disk storage module in the read data mode of operation and read the first four data words from the disk.

Data-service interruption requests occur at 40-microsecond intervals. Four read data commands must empty the disk file data buffer within this interval to avoid a data overrun condition. If the data buffer is not emptied within 40 microseconds, the overrun error bit (bit 2) is set in the interrupt status word and a device-end interruption is presented. Once the data buffer is emptied, the disk storage module is busy to any more read data commands until the next data-service interruption request occurs.

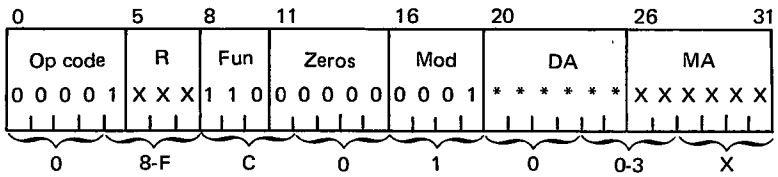
After 128 data words and the check character for the entire data field are successfully read, the end-of-sector bit (bit 7) is set in the interrupt status word and an interruption request is presented.

Programming Note: The terminate command, instead of the four read data or write data commands, must be issued following a data-service interruption request.

Condition code errors to read or write operations can be interrogated to determine if an overrun interrupt is pending or if no interrupt should be expected. If condition code 1 is received, the DSW can be read to determine if the error was command reject or data check. If command reject, no other interruption will be presented. If data check, the program will receive an overrun interrupt. If condition code 2 is received, it indicates that there is another interruption pending. This interruption may indicate overrun or any other error condition, such as, drive became not ready or equipment check.

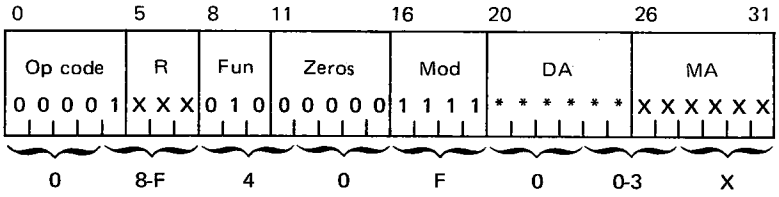
Looping on a halt I/O command before the file becomes ready will keep the file from becoming ready. If a halt I/O or system reset occurs, a load next sector ID or seek operation must be performed to select a head before doing a read, write, read verify or format track operation. If a halt I/O is performed when the file is not ready due to an unsafe condition, the file will become ready. If the file is prepared to interrupt, it will present an interrupt with the attention bit (0) on.

Read Cycle Steal (Cycle Steal Feature)



This command causes the device to cycle steal data from the disk into main storage. The index register (R), or the accumulator if R=000, contains the starting address of the DCB. The DCB contains the information needed to perform the operation. The disk module is busy to all commands, except status reads, from the time this command is issued until the channel end interruption.

Read Residual Address (Cycle Steal Feature)



This command reads the contents of the data address register which is updated by the 5022 attachment subchannel. The purpose of this command is to indicate to the program, for recovery purposes, the exact point during data transfer at which a subchannel error or exception condition occurred. The data address register in the subchannel is not reset by this command or by a halt I/O command. It is reset by a system reset. The read residual address command does not cause an interruption.

If a check character error for the sector ID is detected, the data address register contains the address associated with the first word of that sector. The any-error bit (bit 3), no-record-found bit (bit 5), channel-end bit (bit 11), device-end bit (bit 13), and the read ISWEX bit (bit 15) are set in the ISW. The file-data-check bit (bit 4) is set in the ISWEX.

During a multisector operation, if the ID word is not the next sequential ID, the data address register contains the address associated with the first word of the new sector. The any-error bit (bit 3), channel-end bit (bit 11), device-end bit (bit 13), and the read-ISWEX bit (bit 15) are set in the ISW. The missing-address-mark bit (bit 1) is set in the ISWEX.

If an error is detected during a data transfer on the internal interface or in storage, the data address register contains the address associated with the word that failed to transfer. The any-error bit (bit 3), channel-end bit (bit 11), device-end bit (bit 13), and the appropriate error bit are set in the ISW.

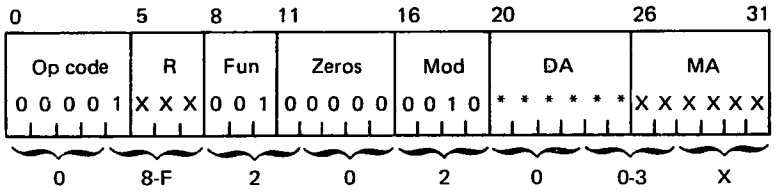
If a check character error for data is detected, the data address register contains an address one greater than that of the last word transferred. The any-error bit (bit 3), channel-end bit (bit 11), device-end bit (bit 13), and the read-ISWEX bit (bit 15) are set in the ISW. The ISWEX contains the file-data-check bit (bit 4).

If an overrun is detected (that is, file data service requirements are not met) the data address register contains the address associated with the word where data service failed. The overrun bit (bit 2), any-error bit (bit 3), channel-end bit (bit 11), and the device-end bit (bit 13) are set in the ISW. (See "Operating and Programming Restrictions" in this chapter.)

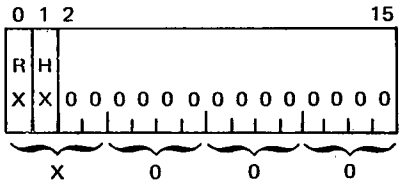
If the track index pulse is encountered before the count becomes zero, the data address register contains an address one greater than the address associated with the last word transferred successfully. The incorrect-length-record bit (bit 6), any-error bit (bit 3), channel-end bit (bit 11), and the device-end bit (bit 13) are set in the ISW.

If the count becomes zero during the data transfer and no errors are detected, a normal end condition occurs. The data address register contains an address one greater than the address of the last word transferred successfully. The channel-end bit (bit 11) and the device-end bit (bit 13) are set in the ISW.

Load Next Sector ID



This command loads into an ID buffer the next sector ID that passes under a selected read/write head. The disk and head are specified by the control information word contained in the index register (R), or the accumulator if R=000. The control information word has the following format:



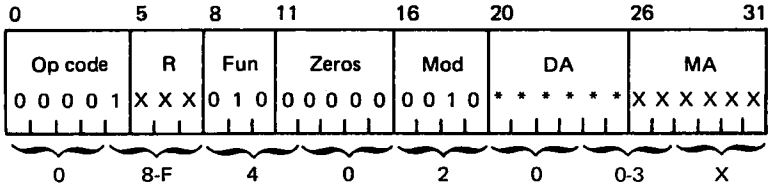
For models 1 and 2 of the disk storage module, the R-bit selects the disk (upper or lower) for use by this and subsequent read or write commands addressed to the module. R=0 selects the upper disk; R=1 selects the lower disk. For models 3 and 4, the R-bit always equals 1.

The head bit (H) selects the read/write head to be used for the selected disk. H=0 selects the upper head; H=1 selects the lower head.

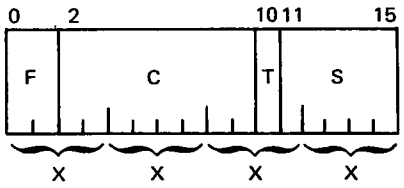
Once a sector ID has been read and checked for the correct ID check character, an interruption request is presented to the processor module. The sector ID must then be read by a read-sector-ID command, described next, in order to obtain the ID just loaded.

Note: The load-next-sector-ID command and the seek command are the only commands that select a disk and/or a read/write head for a subsequent read or write command.

Read Sector ID

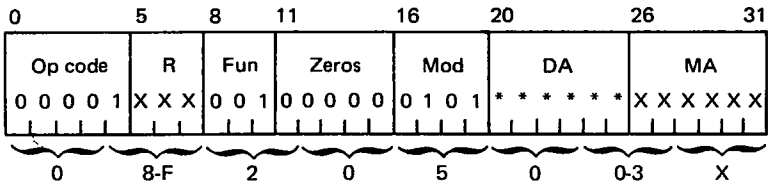


This command transfers a 16-bit sector identifier into an index register (R), or the accumulator if R=000. The read-sector-ID command must be preceded by a load-next-sector-ID command, described previously. The 16-bit identification word transferred to the register has the following format:

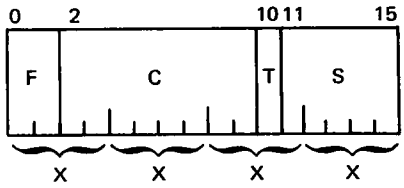


No interruption requests result from the execution of this command.

Read Verify

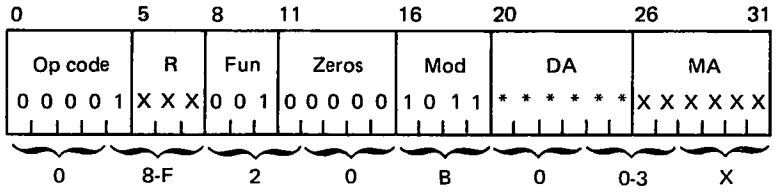


This command reads and verifies the entire data field and its corresponding check character in a sector specified by the control information word contained in the index register (R), or the accumulator if R=000. The contents of the register have the following format:



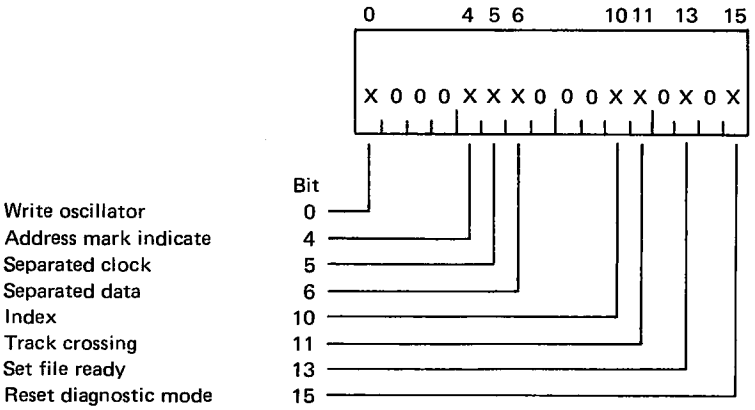
None of the data read is transferred to the processor module; the data field is merely read and the generated check character is compared against the check character read from the disk for that sector. If the check characters are identical, a normal device-end interruption request is presented to the processor module with the end-of-sector ISW bit (bit 7) also set. If the check characters are not identical, the file-data-check bit (bit 4) is set in the interrupt status word and an interruption request is presented.

Write Diagnostic Control



This command disconnects the disk file from its controlling circuitry in the disk storage module so that certain signals and timing pulses can be simulated for testing by the customer engineer.

The control information word used by this command is contained in the index register (R), or the accumulator if R=000, and has the following format:

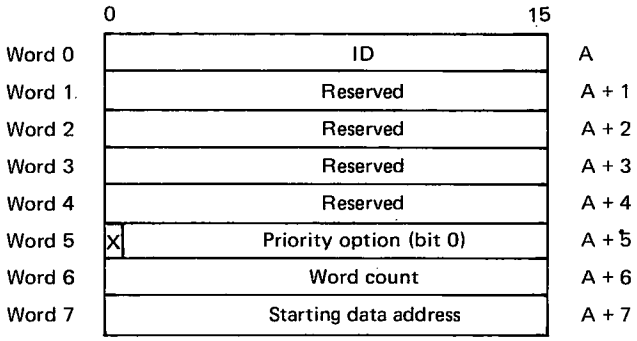


The write diagnostic control command is used in conjunction with the read ISWEX command, described later.

Before signals and timing pulses can be simulated with this command, the disk file must be placed in diagnostic mode by executing a write-diagnostic-control command with a control information word containing 0. A bit in the ISWEX turns on and remains on to indicate that the disk storage module is in diagnostic mode. The file can enter diagnostic mode only if it is not in any other mode at that time. While the file is in diagnostic mode, all other read and write commands can be issued to the disk storage module just as if the module were not in diagnostic mode.

Device Control Block (DCB)

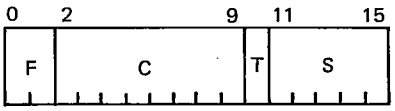
The DCB is an eight word area in storage set up by the user's program and contains the information needed to perform a read cycle steal or write cycle steal operation. The format of the DCB is as follows:



A—the starting address in storage of the DCB. This address is the contents of the index register (R), or the accumulator if R=000, specified in the read cycle steal or write cycle steal command.

ID

The ID word has the following format:



All fields are the same as for write initialize or read initialize.

Priority Option

If word 5 bit 0 is on at DCB fetch time, a higher priority will be assigned to the device at the interface multiplexer.

Word Count

The word count is the number of words to be transferred during the operation. If the word count is 0 when the DCB is fetched, a normal end interruption occurs immediately and the operation is suppressed. The word count is fetched by the subchannel in the 5022 attachment and is decremented by 1 for each word transferred until an interruption condition is met. If the data transfer continues successfully until the count equals 0, an interruption occurs indicating the successful end of the operation. If the word count goes to 0 before the end of a sector:

Read cycle steal–The remainder of the sector is read (but no data is transferred) and the check character is verified prior to the end interruption.

Write cycle steal–The remainder of the sector is set to all 0's and the appropriate check character is accumulated and written at the end of the sector prior to the end interruption.

Data Address

The data address is the address of the first word of data to be transferred. This parameter is fetched by the 5022 subchannel which holds it in a register, updates it, and presents it to the channel during data transfer. On a read operation, the first word from the sector specified in the DCB is stored at this address, the next word from that sector is stored at the next sequential address and so on until the read operation ends. On a write operation, the first word in the data table (pointed to by the data address field) is the first word written onto the sector specified in the DCB until the operation ends.

The read or write operation continues, crossing over sector boundaries, until a termination condition is met. As each new sector is detected during the operation, its ID is checked to ensure that it is, indeed, the next sequential sector.

Error and Status Conditions

Errors detected by the disk storage module are indicated to the controlling program by a condition code and by status words. Status words also indicate general operating conditions of the disk storage module. Figure 15-7 summarizes the definitions of bits in the ISW and ISWEX.

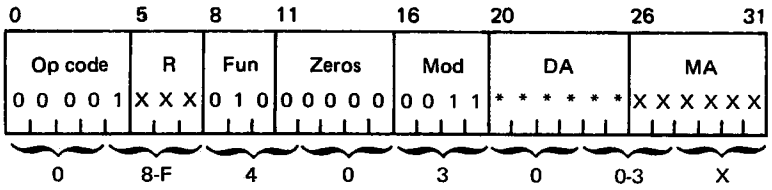
Bit	Interrupt status word (ISW)	Interrupt status word extension (ISWEX)	
		Normal position	Diagnostic position
0	Attention (drive ready)*	File data unsafe*	File data unsafe*
1		Missing address mark*	Missing address mark*
2	Overrun*	Equipment check*	Equipment check*
3	Any error*	Seek check*	Seek check*
4	Drive became not ready*	File data check*	File data check*
5	No record found*	File sense 1	Head settling time
6	Incorrect length record*	File sense 2	Address mark time
7	End of sector*	File sense 3	Pre-ID time
8	Storage data check*		Bit ring zero
9	Program check*	Parallel parity check	Write data to file
10		Check counter error	CC register position 17
11	Channel end	Write gate check	Bit ring inhibit
12	Device busy		Index time 4
13	Device end	Cylinder zero	Data buffer busy
14	Interface data check*	Diagnostic mode	Index sensed
15	Read ISWEX*	Write inhibited	IPL select lower

*The summary status indicator is also set when the resulting interruption request is presented.

Note. ISW bits 6, 8, 9, 11, and 14 are used only with the CS feature. ISW bit 7 is used only with DPC.

Figure 15-7. Summary of bits in ISW and ISWEX

Read ISW



This command stores the disk storage module ISW into the index register (R), or the accumulator if R=000.

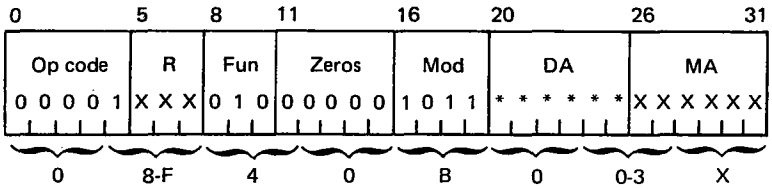
Interrupt Status Word (ISW): Bits in the ISW are set by conditions that occur after a read, write, or seek command is completed, but while the disk storage module is still busy. Since these errors cannot be indicated to the program by the condition code, interruptions result from setting on any ISW bit except bit 12, the busy bit. After the interruption is accepted, executing any command addressed to this module (except read DSW or read ISWEX) resets the ISW.

Descriptions of significant bits in the disk storage module ISW and their meanings follow. An asterisk (*) next to the bit number indicates that the summary status indicator is also set when the resulting interruption request is presented to the processor.

Significant Bits	*Meaning*
0*	Attention. One of the following conditions has occurred: 1. The disk storage module was prepared for interruption requests and the disk file drive became ready after a file start sequence. 2. The disk file drive was ready and the disk storage module has just been prepared for interruption requests.
2*	Overrun. One of the following conditions has occurred: 1. DPC file—A data service interruption request was not serviced within 40 microseconds. The any-error bit (bit 3) and the device-end bit (bit 13) are also set in the interrupt status word by this error condition. 2. CS file—The channel has failed to respond in time to a request for service from the module. Overrun is an exception not an error condition; however, an overrun condition will result in the operation being terminated and an interruption being presented. Any-error bit (bit 3), channel-end bit (bit 11), and device-end bit (bit 13) will be on in the ISW.
3*	Any error. One or more of the ISW error bits 2, 4 through 7, and/or bit 15 are set on.
4*	Drive became not ready. A disk file operation was in progress and the drive became not ready due to a failure in the disk storage module. Additional information can be obtained by reading the ISWEX. The any-error bit (bit 3) and the device-end bit (bit 13) are also set in the interrupt status word by this error condition.

Significant Bits	*Meaning*
5*	No record found. One of the following error conditions has occurred: 1. None of the sector IDs on the selected track compared equally with the sector ID specified by the command after at least one complete revolution of the disk. 2. A specified sector ID was found, but the ID check character generated did not compare equally with the check character read from the sector ID field on the disk. The any-error bit (bit 3) and the device-end bit (bit 13) are also set in the interrupt status word by this error condition.
6*	Incorrect length record. This error bit is set in a cycle steal operation when the subchannel detects a track index pulse and the word count is not 0. Bits 3, 11, and 13 are also set in the ISW.
7*	End of sector. The data field check character has been either read or written, thus signifying the end of the sector. This is not an error condition, so the any-error bit (bit 3) is not set along with this status indicator bit. This indicator does, however, set the device-end bit (bit 13). *Note:* This indicator is not set during a format track operation.
8*	Storage data check. This error bit is set on during output cycle steal operations only. It indicates that the storage location accessed during the current operation contains bad parity and, therefore, the contents are not transferred to disk. The parity in main storage is not corrected. Bits 3, 11, and 13 are also set on in the ISW.
9*	Program check. This error bit is set when the main storage address presented by the device for a cycle steal access is greater than the storage configured on the system. Bits 3, 11, and 13 are also set on in the ISW.
11	Channel end. This bit is set on as the result of termination of a subchannel busy condition. The condition lasts from after the start of a successful start I/O command to the point where the end condition occurs, due either to a normal end condition or to an error end condition. It is always accompanied by the device-end bit in the ISW.
12	Device busy. The disk storage module is performing an operation. This is the only ISW bit that does not cause an interruption request and is not reset by the read ISW command.
13	Device end. The disk storage module has terminated the current operation. The termination may or may not be caused by a detected error condition. This should be the only bit set in the ISW for an error-free data-service interruption request during a read or write operation.
14*	Interface data check. A parity error was detected on a cycle steal interface data transfer. Bits 3, 11, and 13 are also set on in the ISW.
15*	Read interrupt status word extension (ISWEX). A hardware failure or unsafe condition occurred in the disk file drive. The failure or unsafe condition can be further investigated by executing a read ISWEX command addressed to the failing disk storage module.

Read ISWEX



This command stores the interrupt status word extension (ISWEX) from the disk storage module into the index register (R), or the accumulator if R=000. No interruption results from execution of this command.

Bits 5 through 15 can represent one of two sets of conditions in the disk file drive when read by the read ISWEX command. A plug connection is made to the normal or diagnostic position inside the disk storage module to determine which set of conditions is read.

The read ISWEX command is used in conjunction with the write diagnostic control command, described previously, when the status bits are connected to the diagnostic position.

Interrupt Status Word Extension (ISWEX): Bits in the ISWEX are set by conditions that occur after a read or write command is completed, but while the disk storage module is still busy. Setting any one of bits 0 through 4 also sets bit 15 in the disk storage module ISW.

Some conditions that set ISWEX bits are dynamic conditions. That is, the indicator is turned on only for the duration of the condition. However, during execution of a read ISWEX command, the indicators will not change even though the condition may change during the command execution. Significant bits in the ISWEX and their meanings are:

Significant Bits	*Meaning*
0	File data unsafe. The disk storage module detected one of the following unsafe conditions and made the file not ready: 1. Write unsafe a. Write was selected and no write transitions were detected. b. Write was selected and both heads were selected. c. Write was not selected and the write current source was turned on. 2. Erase unsafe a. Write was selected and the erase current source was not turned on. b. Write was not selected and the erase current source was turned on. 3. Read/write selection unsafe a. Read was selected and either write or erase were selected. b. Access mechanism was moving when either write or erase was selected.
1	Missing address mark. The index mark was passed twice and a specified address mark was not found. In this case, ISW bit 5 (no record found) will also be set. If the next sequential ID read from the disk does not compare with the expected ID during a multisector cycle steal operation, missing address mark is set.
2	Equipment check. The disk storage module has detected a failure within the module. Additional information will normally be contained in bits 5 to 15 of this status word.

Significant Bits	*Meaning*
3	Seek check. One of the following error conditions has occurred during a seek operation: 1. The access mechanism, while moving in the forward direction, attempted to move beyond the innermost cylinder on the disk (this is sometimes called access overrun). 2. The access mechanism, while moving in the reverse direction, reached cylinder 0 before the operation was supposed to end. (This condition will not occur during a recalibrate operation.) 3. The access mechanism attempted to move in either the forward or reverse direction for 1 to 2 seconds. 4. A track-crossing pulse was detected at the end of head settling time, indicating that the access mechanism moved past the desired cylinder. The seek-check indicator bit also sets the any-error bit and device-end bit in the interrupt status word.
4	File data check. The check character read from the sector and the check character generated while reading either the sector ID or the data field did not compare equally. The any-error bit (bit 3) and the device-end bit (bit 13) are also set in the interrupt status word by this error condition.

ISWEX Normal Position: Bits 5 through 15 assume the following meaning when the plug is connected to the normal position in the disk storage module.

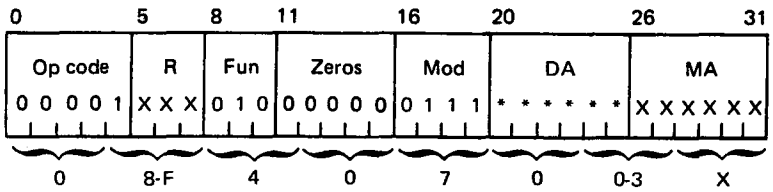
5	File sense 1. This is a pluggable line that can be used to monitor conditions in the disk storage module. The normal position is to monitor the write unsafe latch.
6	File sense 2. This is a pluggable line that can be used to monitor conditions in the disk storage module. The normal position is to monitor the read/write unsafe latch.
7	File sense 3. This is a pluggable line that can be used to monitor conditions in the disk storage module. The normal position is to monitor the erase unsafe latch.
9	Parallel parity check. One of the following conditions has occurred: 1. On a write operation, the parity of the word transferred from the processor did not compare with the parity generated as the word was serialized by the serdes register. 2. On a read operation, the parity generated as the word was deserialized did not compare with the parity of the word when it was transferred to the processor.
10	Check counter error. The contents of the bit count appendage (BCA) counter did not contain the correct value after being reset.
11	Write gate check. The write gate line was active when an index mark was detected and no format track operation was in progress.
13	Cylinder zero. The access mechanism is located at the cylinder 0 position.
14	Diagnostic mode. The disk storage module is in the diagnostic mode of operation.
15	Write inhibited. The file write switch on the disk storage module is in the write inhibit position.

Significant Bits	*Meaning*

ISWEX Diagnostic Position: Bits 5 through 15 assume the following meanings when the plug is connected to the diagnostic position in the disk storage module.

5	Head settling. The read/write heads are settling following a seek operation.
6	Address mark time. The disk file is in address mark (AM) time.
7	Pre-ID time. The disk file is in pre-ID time.
8	Bit ring 0. The bit ring 0 line is active in the disk storage module.
9	Write data to file. A logical 1 is being written on the disk.
10	CC register position 17. Position 17 of the cyclic check register is active.
11	Bit ring inhibit. The bit ring inhibit line is active.
12	Index time 4. The disk file is in index time 4.
13	Data buffer busy. The file is transferring data between the data buffer and a disk.
14	Index sensed. The index on the selected disk is being sensed.
15	IPL select lower. The IPL select switch on the disk storage module was set to address the lower disk for an IPL operation.

Read DSW



This command stores the device status word (DSW) associated with the disk storage module into the index register (R), or the accumulator if R=000.

Execution of the read DSW command resets the bits in the DSW.

Device Status Word (DSW): The bits in the 16-bit DSW are set by error conditions detected during the execution of an immediate command addressed to the I/O module. The module notifies the operating program of these error conditions by returning condition code 1 to the processor. If other commands are attempted before resetting the DSW bits, a condition code of 1 continues to be returned to the program.

The significant bits in the disk storage module DSW and their meanings are:

Significant Bits	*Meaning*
1	Command reject. The disk storage module cannot execute the command for one of the following reasons: 1. The command is invalid for the disk storage module. 2. The present state of the disk storage module prevents executing the command, although it is a valid command for the module. (For example, a read data command issued when the module is in the write data mode.)
4	Intervention required. The disk file drive is not ready for use by the system. This condition may or may not be caused by a failure in the disk storage module. Additional information can be obtained from the ISW and ISWEX.
14	Data check. The disk storage module has detected a parity error involving data. The operation that caused this condition usually can be retried successfully if the error condition is intermittent.

The condition of a track surface is indicated to the disk storage module, and thus to the controlling program, by track flags recorded as part of every sector identifier on a track. The track flags are two indicator bits that are set to represent one of four possible surface conditions: good original, defective original, good alternate, and defective alternate.

Cylinders 1 to 3 are used for alternate tracks when a track in cylinders 4 through 202 is found to be defective. Figure 15-8 shows the correlation between sector identifier bytes recorded on the disk and the control information word used by read and write commands.

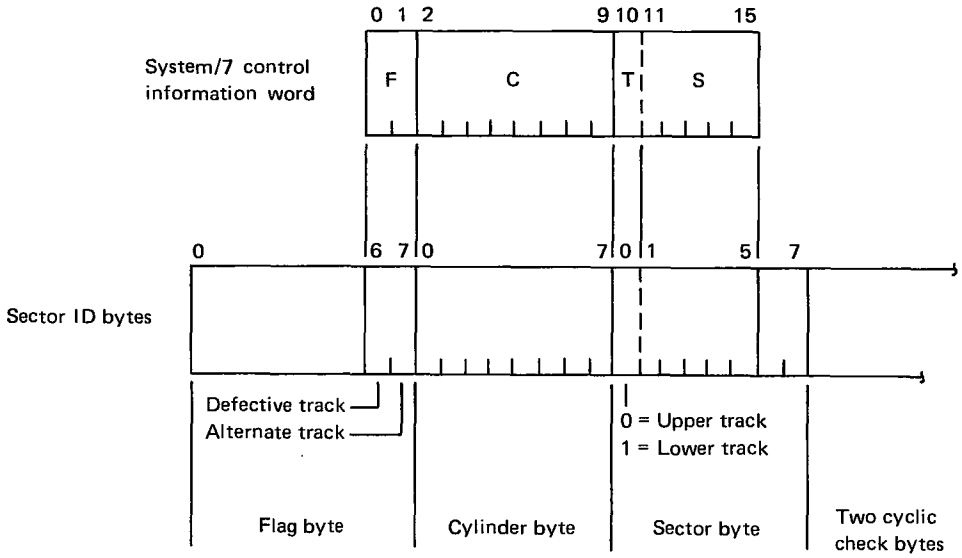


Figure 15–8. Correlation between System/7 word in storage and sector ID on disk

Bits 6 and 7 in the flag byte of the sector identifier in each sector are used to indicate the surface condition of the track in which the sector resides. Bit 6 alone indicates that the track is defective, and bit 7 alone indicates that the track is an alternate. When both bits equal 0, the track is an original "good" track. Both bits set to 1 indicates a defective alternate track. A defective track in cylinder 0 requires disk repair if used for IPL data.

When a track with a bad spot is discovered, it must be marked defective and the program must assign an alternate track to replace the entire defective track. A format track operation must first be performed on the assigned alternate track to identify it as an alternate. The formatting operation writes sector identifiers with flag bits 6 and 7 equal to 0 and 1, respectively, and with C and S bytes containing the identifiers from the defective track. Then, any recoverable data from the defective track must be written on the corresponding sectors of the alternate track. Finally, the defective track must be reformatted to identify it as a defective track. This reformatting operation writes sector identifiers with flag bits 6 and 7 equal to 1 and 0, respectively, and with C and S bytes containing the identifiers from the alternate track.

Should an alternate track be defective, it should be reformatted to identify it as defective. This is done by writing sector identifiers with flag bits 6 and 7 both equal to 1, and with C and S bytes containing the same identifiers that they originally held.

The track identifier fields are summarized as follows:

Good: Bits 6 and 7 of the F byte are both 0. The C and S bytes contain the cylinder, track, and sector addresses that are correct for that track.

Defective: Bit 6 is 1 and bit 7 is 0 in the F byte. The C and S bytes contain the cylinder, track, and sector addresses from the alternate track.

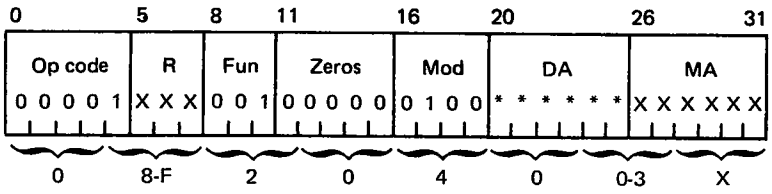
Good alternate: Bit 6 is 0 and bit 7 is 1 in the F byte. The C and S bytes contain the cylinder, track, and sector addresses from the defective track.

Defective alternate: Bits 6 and 7 of the F byte are both 1. The C and S bytes contain the cylinder, track, and sector addresses originally assigned to them.

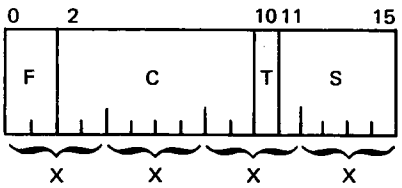
TRACK INITIALIZATION

Before any disk can be used, the surfaces must first be formatted with address marks, sector identifiers, and gaps. A special write command records these fields on a disk surface. Once a disk is formatted, the remaining disk I/O commands can be used. The procedures recommended for formatting a disk are described later under "Track Initialization Procedures."

Format Track



Address marks, sector identifiers, gaps, and 0's in the data fields are written on a track. (The read/write head used to write the track must be selected by a previous seek command or load-next-sector-ID command.) Sector identifier data to be written in the first sector on the track is contained in the index register (R), or the accumulator if R=000. The data word has the following format:



The track bit (T) is an extension of the sector address. Thus, sector numbers 0 through 23 are written on the upper track and sector numbers 32 through 55 are written on the lower track.

The cylinder number field (C) must contain the physical number, in binary, of the cylinder (0000 0000 to 1100 1011). This number is the same for every sector ID on the same track.

The sector number field (S) must be set to 0. This number is incremented automatically for each successive sector formatted on a track.

Formatting starts at the first sector past the index mark and continues for 24 sectors. For a previously unformatted disk, the flag bits (F) in the data word are set to 0. A device-end interruption request is presented after the track is completely formatted. After the track has been formatted, a read verify operation must be performed to check for any surface defects on the track. If an error occurs during the verify operation and the track is determined to be defective, the track must be reformatted as described previously under "Flagging."

Note: The disk storage module gives a busy indication to all other commands while a track is being formatted.

Track Initialization Procedures

The following procedures are recommended for track initialization programs used with the 5022 Disk Storage Module. They must be performed for each disk (upper or lower) to format the tracks and analyze the condition of the disk surface. (Steps 2 to 5 must be performed twice for each cylinder—once for the upper track and once for the lower track.)

	Operation	*Description*
1.	Read sector ID	Determine that the track has not been previously flagged. (This step need not be performed when initializing a previously unused disk.)
2.	Format track	Write address marks, sector identifiers, and gaps on the track.
3.	Read verify	Check all the sectors to ensure that data can be recovered correctly. If an error occurs, go to step 10.
4.	Write data	Write on each sector with a data word of hexadecimal 5555.
5.	Read verify	Check all the sectors to ensure that data can be recovered correctly. If an error occurs, go to step 10.
6.	Proceed	If both tracks in the cylinder are formatted, seek to the next cylinder and repeat steps 2 through 5. If both tracks are not formatted, repeat steps 2 through 5 for the other track in this cylinder.
7.	Repeat	Perform steps 2 through 6 until all tracks have been processed.
8.	Read sector ID	Read a sector ID on all tracks to check for seek errors. A seek error during a format track operation causes two different tracks to contain the same identifiers or the identifiers for one track to be missing. If a seek error occurred during the formatting operation, repeat steps 1 through 7.
9.	Proceed	Perform steps 2 through 8 at least once to complete the initialization procedure.
10.	Error handling	If an error occurs, analyze the ISW and ISWEX. If a missing address mark or file data check occurs, retry a read verify command at least 10 times. On the first unsuccessful retry that indicates a missing address mark or file data check, flag the track as defective and go to step 11. If all 10 retries are successful, proceed with the initialization procedure from the point at which it was interrupted. For any error other than missing address mark or file data check, follow the normal error recovery procedures described later under "Suggested Error Recovery Procedures."
11.	Assign track	Alternate tracks must be proved reliable by steps 2 through 5 before they are used as alternates.
12.	Format track	Write sector identifiers on the defective track with the flag bits set to 1 and 0, respectively, and the cylinder and sector values of the alternate track. A defective alternate track should be formatted with its own address and flag bits both set to 1.
13.	Read sector ID	Read a sector ID from the defective track just formatted. If the address of the alternate track cannot be recovered, the disk must be repaired unless this is an alternate track.

Operation		*Description*
14.	Seek	Seek to the alternate track.
15.	Format track	Write sector identifiers on the alternate track with the flag bits set to 0 and 1, respectively, and the same cylinder and sector values that were originally in the defective track.
16.	Continue	Proceed with initialization on the next track. (Return to step 7 or step 10.)

The basic requirement is for one pass through steps 1 to 8. An option must be provided to allow additional testing so that a track can be positively identified as good or defective.

After initialization, only two types of programs should change track flags: (1) initialization programs, which must be able to ignore all previously flagged tracks and unconditionally flag or unflag any individual track; and (2) operating programs that can dynamically flag a track must perform steps 11 through 15 of this procedure.

SUGGESTED RECOVERY PROCEDURES

Error or exception conditions are indicated by bits that are set in the device status word (DSW), the interrupt status word (ISW), and the interrupt status word extension (ISWEX) for the disk storage module. Minimum error recovery procedures are defined for the disk storage module in the following discussions.

Recovery from DSW Errors

A condition that sets DSW bit 4 can be checked for recovery by first reading the ISWEX and checking for the presence of the unsafe bit (bit 0). If no unsafe condition exists (bit 0 = 0), the disk file drive has not completed its starting sequence.

If an unsafe condition does exist, a halt I/O command should be issued to reset the conditions and the original operation retried. If the unsafe condition cannot be reset after five attempts, operator intervention is required.

Recovery procedures for DSW bits 1 and 14 are described in Chapter 4 under "Device Status Word," since these status conditions are defined for all I/O modules.

Recovery from ISW Error or Exception

Bit 2—Overrun

If the overrun condition occurred while executing a write sequence, any pertinent parameters (such as word count) should be saved and the original sequence of operations retried. After eight unsuccessful retries, the user should be notified.

If the overrun condition occurred while executing a read sequence, any pertinent parameters (such as word count) should be saved and the original sequence of operations retried. After 16 unsuccessful retries, the user should be notified.

Bit 4—Drive Became Not Ready

It should be determined if an unsafe condition exists (ISWEX bit 0 present). If this condition does exist, a halt I/O command should be issued to reset the unsafe condition(s) and the original operation retried. After five unsuccessful retries, the user should be notified. If no unsafe condition exists, the user should be notified to restart the file.

Bit 5—No Record Found

It should be determined if the file-data-check error bit (ISWEX bit 4) is also present. If it is present, any pertinent parameters (such as sector ID) should be saved and the operation retried. After 16 unsuccessful retries, the user should be notified.

If the file-data-check error bit is not present, it should be determined if the missing address mark indicator (ISWEX bit 1) is present. If it is present, a recalibrate operation should be performed and the original sequence of operations retried. After 16 unsuccessful retries, the user should be notified.

Verify that the heads are positioned over the desired track by executing the load and read next sector ID sequence. Retry the original operation and notify the user after 16 unsuccessful retries.

Bit 6—Incorrect Length Record

This condition may not be treated as an error by the system program, in which case no recovery procedure is recommended. If this condition is considered an error, however, the suggested procedure is to validate the DCB and retry the operation.

Bit 8—Storage Data Check

Read the residual address from the subchannel address register. The command to do this is an immediate read with a modifier of hex F (1111). The command can be used by the program to determine the contents of the data address register which is held and updated by the 5022 subchannel. Use this address to determine what storage location had bad parity and in which sector the error occurred. Validate that location and retry the operation beginning with the sector where the error occurred. Another procedure would be to validate the entire data table in storage and retry the operation in its entirety.

Bit 9—Program Check

Validate the address of the DCB given on the start I/O command. If correct, validate the DCB by correcting the word count and/or the data address fields and retry the operation.

Bit 14—Interface Data Check

Read the residual address to determine in which sector the error occurred. Validate the DCB and retry the operation five times. If unsuccessful, issue a halt I/O command to the disk module, to reset the system, and retry once. If unsuccessful, assume that a hard failure condition exists.

Bit 15—Read ISWEX

Any necessary parameters should be saved and the interrupt status word extension (ISWEX) should be read. If the ISWEX indicates a hardware error (bit 2 present), a halt I/O command should be issued to the module and then the original sequence of operations retried. After one unsuccessful retry, the user should be notified.

Recovery from ISWEX Errors

Bit 0—File Data Unsafe

A halt I/O command should be issued to the module and the original operation retried. After five unsuccessful retries, the user should be notified.

Bit 1—Missing Address Mark

It should be verified that the disk has been initialized by reading the next sector ID on a different track or cylinder. If the disk has been initialized, an attempt should be made to restore the address mark by reformatting the original track with its original identifiers. Then, a sector ID should be read from the reformatted track. If unsuccessful, the user should be notified that the track is defective.

Bit 2—Equipment Check

A halt I/O command should be issued to the module and the original sequence of operations retried. After five unsuccessful retries, the user should be notified.

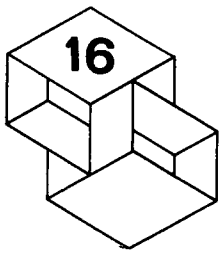
Bit 3—Seek Check

A recalibrate operation (seek in the reverse direction with N ⩾ 224) should be performed and the original seek operation retried. After 16 unsuccessful retries, the user should be notified.

Bit 4—File Data Check

It should be determined if the no-record-found bit (ISW bit 5) is also present. If it is present, any pertinent parameters (such as sector ID) should be saved and the operation retried. After 16 unsuccessful retries, the user should be notified. If the no-record-found bit is not present, the original sequence of operations should be retried. After 16 unsuccessful retries, the user should be notified.

CHAPTER CONTENTS



The IBM 5010 Processor Module Model Bxx attaches the System/7 directly to an IBM 1130 Computing System via the 1130 Storage Access Channel (SAC) and the 1130 host attachment in the System/7. The 1130 host attachment in the processor module permits direct, two-way storage-to-storage communication and information transfer between the two systems. The 1130 host attachment precludes using the System/7 as a satellite processor by means of the Asynchronous Communications Control Attachment or the Binary Synchronous Communications Adapter.

FUNCTIONAL DESCRIPTION

The 1130 host attachment can be connected to either the 1130 SAC or the 1130 SAC II. The host attachment transfers data or instructions at the 1130 storage speed (454 kHz on the 2.2-microsecond 1130 or 277 kHz on the 3.6-microsecond 1130). Although the attachment requests 1130 cycles so quickly that it may prevent instruction execution in the 1130 processor (except for interruption level 0 or 1), the System/7 storage speed permits execution of System/7 instructions during a data transfer.

Interruption level 0 or 1 instructions are executed in the 1130 during a data transfer because the System/7 interrupts the 1130 on interrupt level 3 and, consequently, the System/7 must obey the inhibit-cycle-steal-request line from the 1130 during higher-priority interruptions. For further details, refer to the manual *1131 Central Processing Unit and 1133 Multiplex Control Enclosure Storage Access Channel–Original Equipment Manufacturer's Information*, Order No. GA26-3645.

Use of System/7 storage by the two systems requires both machine and program controls to resolve conflicts for access to System/7 storage. The System/7 has an interruption mechanism that enables either system to alert the other for service. To determine the nature of the service required, programming must maintain current control parameters in a System/7 storage area.

All data transfers are initiated by the 1130, and all end or error conditions are signaled to the 1130 by means of interruptions. No direct communication exists between the 1130 and the System/7 I/O modules; data transfers are strictly storage-to-storage.

Although no provision is made for parity within the 1130 SAC, the 1130 host attachment generates the necessary parity for data destined for System/7 storage.

A host attachment switch located on the System/7 console controls the connection between the System/7 and the 1130. When in the disable position, interrupt and cycle-steal requests cannot go to the 1130, all 1130 instructions are ignored, and condition code 3 is returned if a System/7 set interrupt command is addressed to the host attachment. When the switch is in the enable or enable and IPL position, the 1130 and System/7 can communicate through the SAC in a normal manner. The 1130 attachment is reset when the 1130 console reset key is depressed or if the System/7 console host attachment switch is moved from the enable or the enable and IPL position. IPL from the 1130 is described later in this chapter under "Electronic Initial Program Load (EIPL)."

Interruptions to 1130

The 1130 host attachment presents interruptions to the 1130 on its interruption level 3. Interruption requests are made to the 1130 under any of the following conditions:

1. The DSW attention bit is set on because a System/7 set interrupt command is directed to the 1130 (if the interruption controls do not inhibit the interruption request).
2. The DSW operation-end bit is set on. (The count=0 DSW bit, the power/thermal warning bit, or an error bit is also on for this interruption request.)
3. The DSW power or thermal warning bit is set on by the corresponding condition (if the interruption controls do not inhibit the interruption request). The DSW ready bit remains on until the power actually fails. If a data transfer operation is in progress when the power/thermal warning bit comes on, the operation-end bit is also set on, requesting an interruption to the 1130.

The interruption that occurs at the operation-end time of a data transfer can be directed to the System/7 as well as to the 1130. However, if an error is detected during the data transfer, the operation is terminated and the interruption is directed to the 1130 only (count=0 DSW bit may or may not be set on).

When the System/7 directs a set interrupt command to the 1130 attachment, the DSW attention bit (bit 0) is set on unless it is already on. If already on, condition code 2 is returned to the System/7. If the 1130 attachment is busy, the attention bit is set on and indicated to the 1130 at the time an operation-end interrupt occurs for the operation in progress. If the 1130 attachment is not busy, the attention bit is set on and an interruption request is made to the 1130 on its priority level 3 unless attention interruptions are inhibited by the host attachment interruption controls.

The host attachment interruption controls, which permit or prevent attention and power/thermal interruptions to the 1130, are determined by the setting of modifier field bits 28 to 30 in an 1130 IOCC. Modifier bits 28 and 29 are the *basic* interruption control; bit 30 is a *temporary* interruption control. Attention and power/thermal interruptions are permitted only if the basic status is "permit" and there is no temporary "prevent." If the basic status prevents attention and power/thermal interruptions, these interruptions are not permitted again until another control IOCC that alters the basic interruption control status is executed to permit the interruptions.

If interruptions are temporarily prevented as directed by bit 30 in a control IOCC, they are not permitted again until one of the following occurs:

1. The DSW is reset.
2. Another control IOCC is executed with bit 30 changing the previous temporary status.
3. An initiate read or initiate write IOCC is executed.

Following any of these events, the 1130 host attachment returns to the basic interruption control status.

Attention and power/thermal interruptions to the 1130 are also prevented by either a System/7 power-on reset or an 1130 Storage Access Channel reset. These interruptions are not permitted again until a control IOCC is executed to establish a basic interruption control status.

Interruptions to System/7

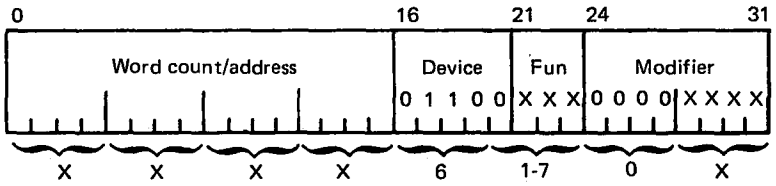
The program in the 1130 can request an interruption to the System/7 processor module. The interrupt priority level, sublevel, and device address are fixed for the 1130 host attachment; therefore, no prepare I/O command is required. As directed by a bit in an initiate read or initiate write IOCC, the request to the System/7 processor is made on priority level 3 with a sublevel of 0. The interruption request presented to the System/7 by the 1130 is handled the same as any other priority interruption request as described in Chapter 3 under "Priority Interruptions." When the processor services the request, the 1130 host attachment presents a device address of 011100.

INSTRUCTIONS AND COMMANDS FOR 1130 SYSTEM

Standard 1130 I/O instructions and control commands are used to program the host attachment from the 1130. An 1130 execute I/O (XIO) instruction is issued to address the appropriate IOCC. The IOCC specifies the operation to be performed and the device to which the operation is directed. For data transfer operations, the IOCC also specifies the word count or address of the data to be transferred. For a general description of the XIO instruction and IOCC's, refer to the manual *IBM 1130 Functional Characteristics*, Order No. GA26-5881.

I/O Control Commands (IOCC)

An IOCC must start at an even storage address in the 1131 processor and has the following format:



The IOCC fields are described as follows:

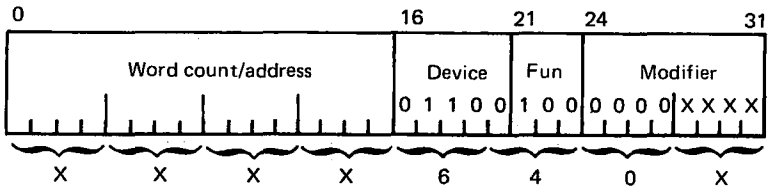
Word Count/Address: This field is used to pass the three values required by the 1130 host attachment in order to perform a data transmission: (1) a transmission word count, (2) a System/7 storage address, and (3) an 1130 storage address. The count specifies the number of data words to be transferred. The addresses establish the beginning of the data tables between which the data transfer is to occur. The count and one of the addresses are transferred to System/7 storage by two separate control IOCC's. The third value, an address, is transferred by the 1130 initiate read or initiate write IOCC.

Device: This 5-bit field identifies the I/O device to which the IOCC is directed. In this case, a binary 01100 is the code identifying the System/7.

Function (Fun): The 3-bit function code determines the specific I/O operation to be performed.

Modifier: This 8-bit field provides additional information, when necessary, for the function specified.

Control



An XIO instruction with a control IOCC loads the 1130 host attachment in the System/7 with one of two parameters necessary before starting a data transfer operation between the System/7 and the 1130. The two parameters are:

1. The System/7 storage address at which the data transfer is to begin.
2. The count of the number of 16-bit data words to be transferred.

The 1130 storage address involved in the data transfer is indicated subsequently in either the 1130 initiate read or initiate write IOCC.

Since only one of the two parameters can appear in a single control IOCC, two control IOCC's are required (and, hence, two XIO instructions) prior to performing the actual data transfer.

Modifier bits 30 and 31 signal which, if either, of the two parameters is contained in the control IOCC as follows:

Bit 30 = 0 – Neither parameter being transferred (ignore bit 31)
Bit 30 = 1 – Perform function specified by bit 31
Bit 31 = 0 – Word count being transferred to attachment
Bit 31 = 1 – System/7 storage address being transferred to attachment

Modifier bits 28 and 29 of the control IOCC determine the basic interruption controls to be established in the attachment. To establish the basic control status, modifier field bits 28 and 29 are set as follows:

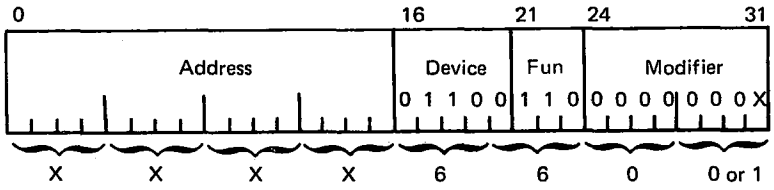
Bit 28 = 0 – Ignore bit 29
Bit 28 = 1 – Perform function specified by bit 29
Bit 29 = 0 – Permit attention and power/thermal interruptions
Bit 29 = 1 – Prevent attention and power/thermal interruptions

Modifier field bit 30 serves a dual purpose in a control IOCC. This bit also establishes a temporary interruption control status as follows:

Bit 30 = 0 – No effect; return to basic control status
Bit 30 = 1 – Prevent attention and power/thermal interruptions

The temporary control status is temporary only in the sense that an initiate read command, an initiate write command, or another control command, immediately following such a control IOCC, can return the interruption control status to the basic control status.

Initiate Read

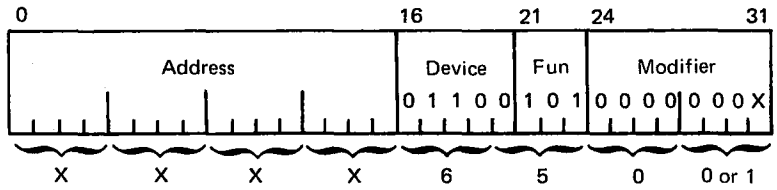


An XIO instruction with an initiate read IOCC sends a block of contiguous data from System/7 storage to 1130 storage. The starting location of the System/7 data and the number of words transferred must have been established by previously executed control IOCC's. The address field of the initiate read IOCC contains the starting location in 1130 storage for the data to be received.

Modifier field bit 31, described previously in "Interruptions To System/7," is used to control interruptions as follows:

Bit 31 = 0 – Do not interrupt System/7 for operation end
Bit 31 = 1 – Interrupt System/7 for operation end

Initiate Write

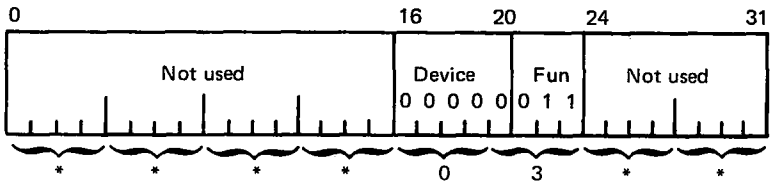


An XIO instruction with an initiate write IOCC sends a block of contiguous data from 1130 storage to System/7 storage. The number of words transferred and the System/7 starting location into which they are stored must have been established by previously executed control IOCC's. The address field of the initiate write IOCC contains the starting location in 1130 storage for the data to be transmitted.

Modifier field bit 31, described previously in "Interruptions To System/7," is used to control interruptions as follows:

Bit 31 = 0 – Do not interrupt System/7 for operation end
Bit 31 = 1 – Interrupt System/7 for operation end

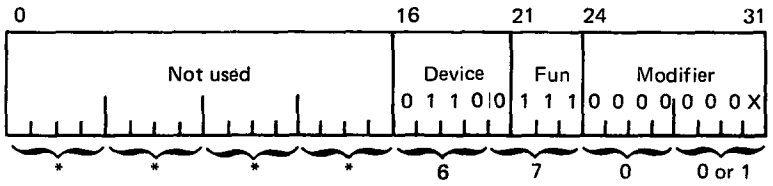
Sense Interrupt



An XIO instruction with a sense interrupt IOCC loads the 1130 accumulator with the interrupt level status word (ILSW) associated with the highest priority level that is on, in order to determine the interrupting device. The sense interrupt command is common to all 1130 I/O devices; therefore, no device-code field is needed.

A System/7 interrupt request to the 1130 sets on ILSW bit 4 for interruption level 3. (The System/7 and the IBM 2250 Display Unit are mutually exclusive on the 1130 SAC.)

Sense Device



An XIO instruction with a sense device IOCC loads the 1130 accumulator with the DSW from the 1130 host attachment in the System/7. The 1130 can determine the cause of an interruption by analyzing these DSW bits.

Modifier field bit 31 has the following meaning in this IOCC:

Bit 31 = 0 — Do not reset DSW bits
Bit 31 = 1 — Reset DSW bits

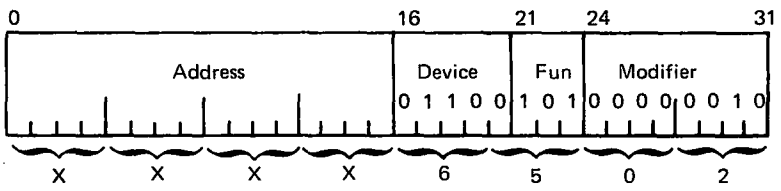
DSW bits can also be reset by turning on System/7 power or by the reset line in the 1130 SAC.

Device Status Word (DSW): The 16-bit device status word associated with the 1130 host attachment has various bits set on to indicate program operating status and detected errors.

The significant bits in the DSW presented to the 1130 processor and their meanings are:

Significant Bits	*Meaning*
0	Attention. A System/7 set interrupt command is directed to the 1130. This is not an error.
1	Operation end. The word count equals zero, or a power/thermal warning or error was detected during a data transfer operation.
2	Invalid storage address. The 1130 has attempted to address a storage location outside the installed capacity of the System/7. This is an error.
3	Data check. A System/7 parity error was detected by the 1130 host attachment when fetching words from System/7 storage. This is an error.
4	Count = 0. The word count register in the 1130 host attachment is equal to 0. This is not an error.
5	Power or thermal warning. A power failure or thermal warning condition has occurred. This bit is not reset by the sense-device IOCC. (Refer to "Power Failure and Thermal Warning" in Chapter 1 and "Class Interruptions" in Chapter 3.)
6	Storage control check. The 1130 host attachment detected that System/7 storage has not responded to the attachment's request for a storage cycle, sometimes referred to as an "overrun" condition. This is an error.
14	Ready. The System/7 is on line and power is good. This bit is not reset by the sense-device IOCC. This is not an error.
15	Busy. The 1130 host attachment is performing a data transfer operation between the attachment and System/7 storage. This bit is turned off as soon as the operation-end bit (bit 1) is set on. Any 1130 command (except sense-device) to the 1130 host attachment is ignored if the busy bit is on. This is not an error.

Electronic Initial Program Load (EIPL)



An XIO instruction with this special initiate-write IOCC is used to IPL the System/7 from the 1130. Modifier bit 30 in this IOCC serves a dual purpose for the EIPL function. This bit also establishes a temporary "prevent" status for attention and power/thermal interruptions. Thus, the 1130 will not recognize an attention or power/thermal warning interruption during an EIPL to the System/7. The basic interruption status must be reestablished by a control command with modifier field bit 30 = 0 or by an initiate read or initiate write command.

The host attachment switch on the System/7 console must be in the enable and IPL position for this command to perform the IPL. When the 1130 host attachment recognizes the EIPL command, the System/7 does a system reset and enters the wait state. The host attachment sets the System/7 address to 0 and proceeds with the EIPL as if it were a normal initiate-write from the 1130.

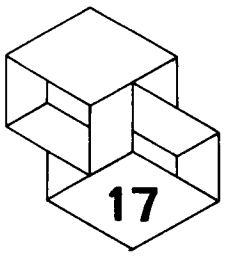
For an error-free termination, the System/7 instruction address register is set to a 0 value, priority level 3 is activated, and System/7 begins to execute instructions starting at location 0. A standard operation-end interrupt is also generated in the 1130.

For an error termination, the System/7 is not informed of the IPL termination, a standard operation-end interrupt is generated in the 1130, and the error condition is indicated by setting a corresponding status bit in the DSW for the 1130.

A word count must be established in the 1130 host attachment prior to attempting an EIPL operation from the 1130. This word count is established by an XIO instruction with a control IOCC as described in this chapter under "Control."

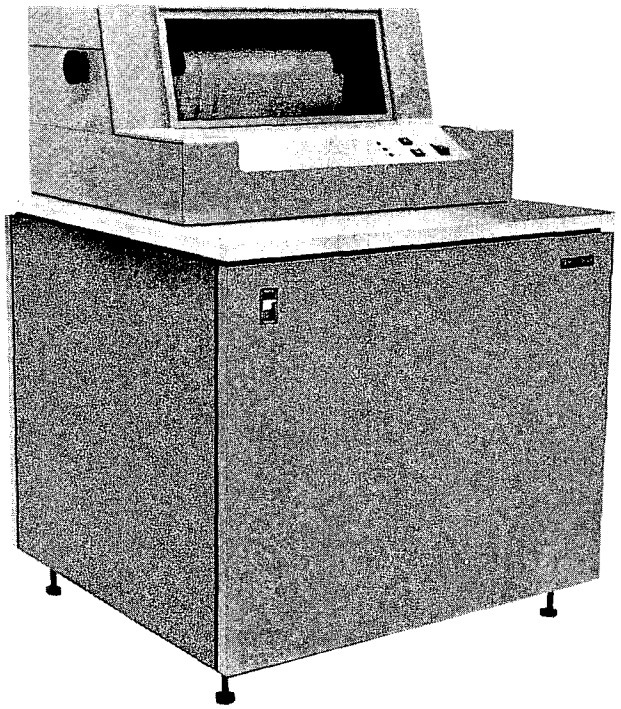
A System/7 address need not be established, since the host attachment sets this address to 0.

CHAPTER CONTENTS



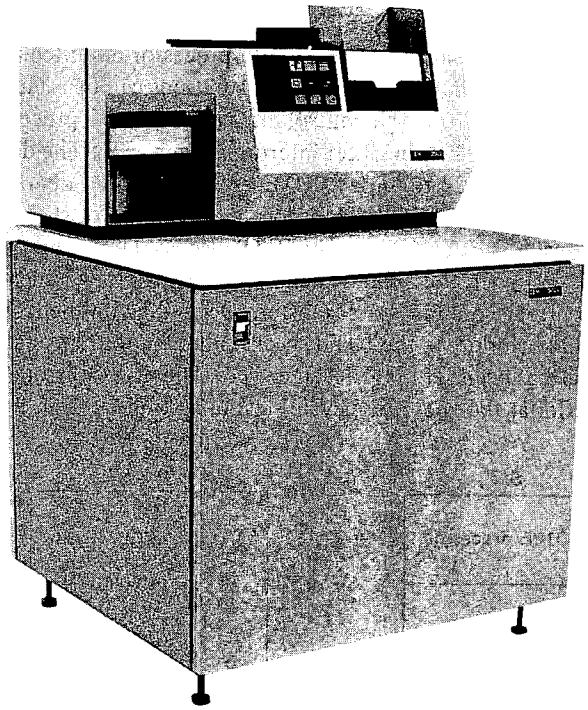
The IBM 5024 I/O Attachment Enclosure provides medium speed line printing and card read capabilities for the System/7 via a Line Printer and a 2502 Model A2 Card Reader. The 5024 differs from conventional System/7 I/O modules in that it resides outside of the 5026 within its own enclosure. The enclosure houses the power and logic necessary to control the printer and card reader mounted on it. Figures 17-1 through 17-3, illustrate the 3 model configurations for the 5024 which are:

- Model 1—Line Printer only.
- Model 2—2502 Card Reader Attachment only.
- Model 3—Line Printer and 2502 Card Reader attachment.



BR2915

Figure 17-1. 5024 Model 1



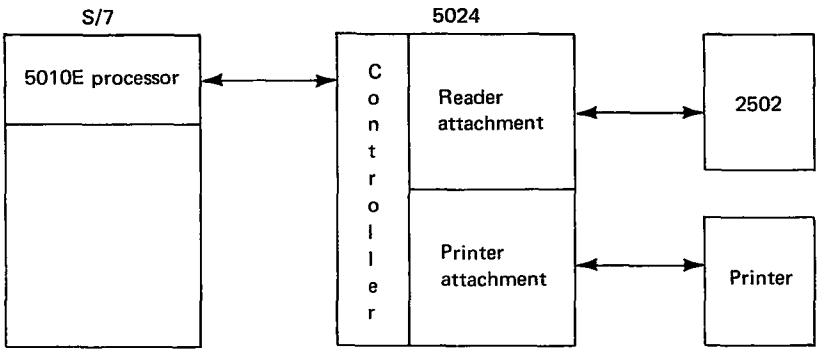
BR2916

Figure 17-2. 5024 Model 2 (with 2502 Card Reader Installed)

BR2917

Figure 17-3. 5024 Model 3 (with 2502 Card Reader Installed)

The 5024 can attach to the 5010E processor mounted in a 5026, A2, C3, or C6 enclosures. Any one of the three model configurations can be attached. The 5024 communicates with the System/7 through control logic in the 5010E processor. The System/7 communicates with the 5024 through set interrupt commands. The data words selected by the commands direct the 5024 to initiate I/O operations or reset operations. The actual command to be performed by the 5024 is located in an area of storage called the Input Output Transfer Block (IOTB). The IOTB also contains other parameters that are used during the I/O operation. Once the initiate I/O has been issued, cycle steal operations are used to accomplish data transfer. Depending on the I/O device selected (either the 2502 or Printer), the IOTB specifies the appropriate status buffer starting address that has been set up for each device. The data controller fills the specified status buffer at the end of the cycle steal operation.



Data Controller

All control logic for the 5024 is in the data controller. A single System/7 XIO instruction (initiate I/O) initiates a series of cycle steal sequences that are terminated with an interrupt from the 5024. The data controller accesses storage location 0008 for the address of the I/O Transfer Block (IOTB) which specifies commands and control information for the 5024.

I/O Transfer Block

Figure 17-4 shows the format of the IOTB and status buffer. The IOTB is a five word block that contains information set up by the user's program to control data transfer between the System/7 and the 5024. A typical data transfer between the System/7 and a device in the 5024 is as follows:

1. The IOTB is established by user program in any available main storage location.
2. Location 0008 is initialized by user program with the address of the IOTB.
3. User's program issues initiate I/O command.
4. The controller cycle steals address from location 0008.
5. The controller cycle steals each word of the IOTB and determines the command, device to be selected (printer or card reader), data starting address, and word count.
6. The controller performs the data transfer between main storage and the selected device via cycle steal.
7. The controller transfers device information to the status buffer in main storage.
8. The controller interrupts System/7 to indicate that operation is complete, and the 5024 is ready for the next transfer.
9. The user program examines status buffer for ending conditions.
10. The user program clears status buffer.

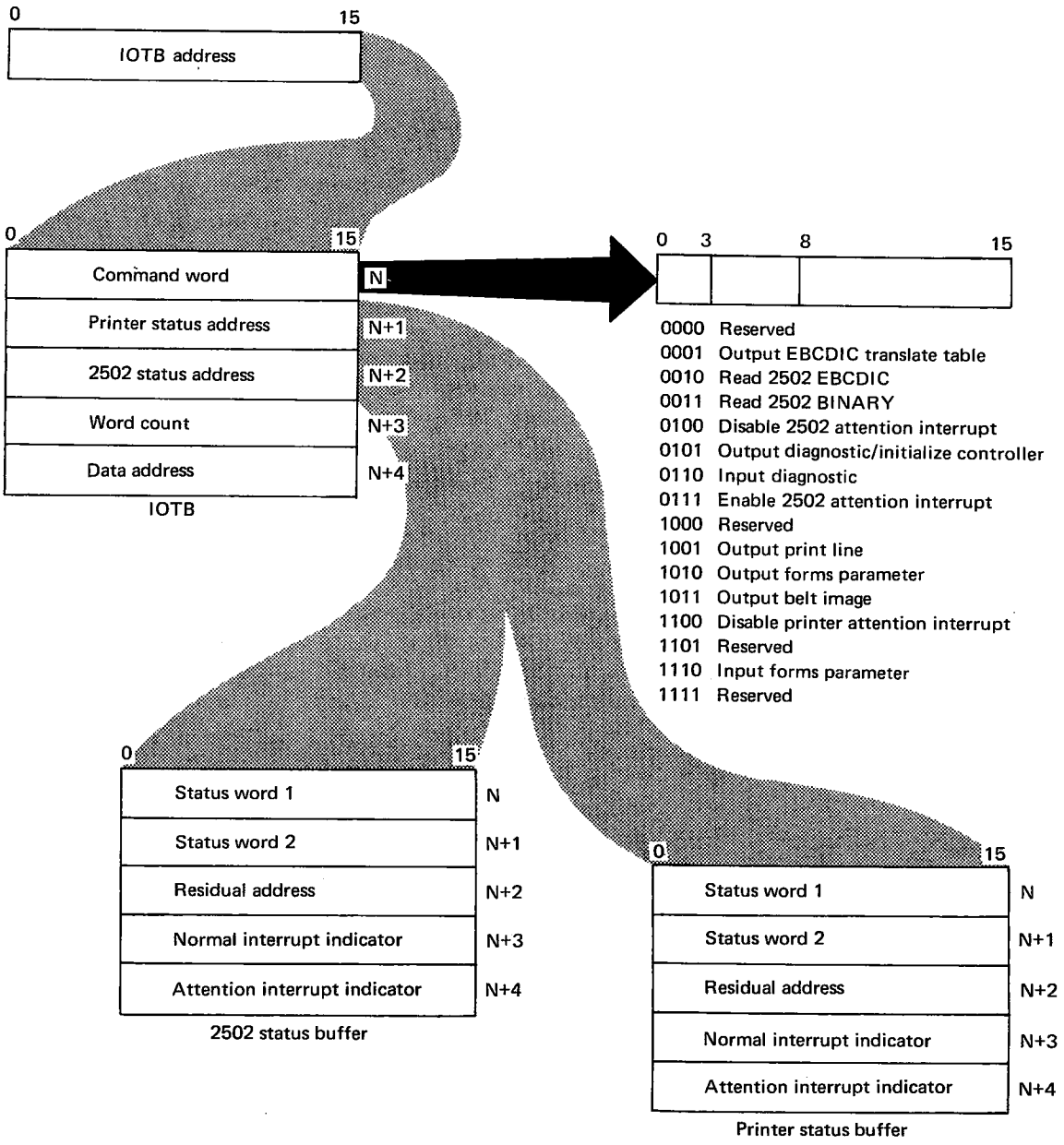


Figure 17-4. IOTB and status buffer format

IOTB Words

The five IOTB words are described as follows:

Word 1–Command Word. The first four bits of this word contain the command to be executed by the controller. The last eight bits contain carriage control information. Carriage control applies to operation of the printer when the output print command is issued. For further definition of carriage control see the command word under Printer.

Word 2–Printer Status Address. The address of a five word status buffer for the Printer.

Word 3–2502 Status Address. The address of a five word status buffer for the 2502.

Word 4–Word Count. The number of words to be transferred during the operation. The output diagnostic/initialize controller, input diagnostic, and output print line commands require the user to set a valid word count. The controller establishes the word count for all other commands.

Word 5–Data Address. The address of the buffer in the System/7 into which data will be stored or transferred.

Status Buffers

The Status Buffer is a five word block in main storage allocated by the user program and filled by the controller with status and control information. The Printer and 2502 each have a status buffer. Words 1 through 4 are stored on a normal interrupt. Word 5 is stored on an attention interrupt.

Word 1 and Word 2. These status words are explained in the 2502 and Printer sections of this chapter under "Status Buffers."

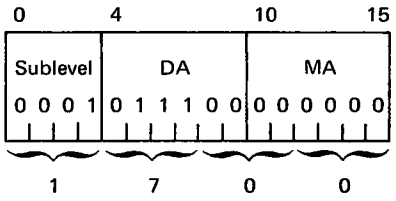
Word 3–Residual Address. The last System/7 address accessed by the controller in a command sequence.

Word 4–Normal Interrupt Indicator. Bit 0 of this word is set to 1 by the controller to indicate that a command sequence ending interrupt has occurred for the corresponding I/O device. This word must be read and set to zero by the System/7 program when it services an interrupt from the 5024. Bits 1 through 15 are set to zero by the controller.

Word 5–Attention Interrupt Indicator. An interrupt will be generated and bit 0 of this word is set to 1 when the associated I/O device goes from a not ready state to a ready condition. Bits 1 through 15 are set to zero by the controller. Status buffer words 1, 2, 3, and 4 are not changed by an attention interrupt. The program must read this word and set it to zero when servicing an interrupt from the 5024.

Interrupt Presentations

Interrupts occur on level 1 sublevel 1 or level 3 sublevel 1. The level at which an interrupt occurs is determined by wiring in the System/7 when the 5024 is installed. This makes the interrupt structure non-programmable and eliminates the use of a prepare I/O instruction. When an interrupt is accepted, the accumulator will have the following configuration:



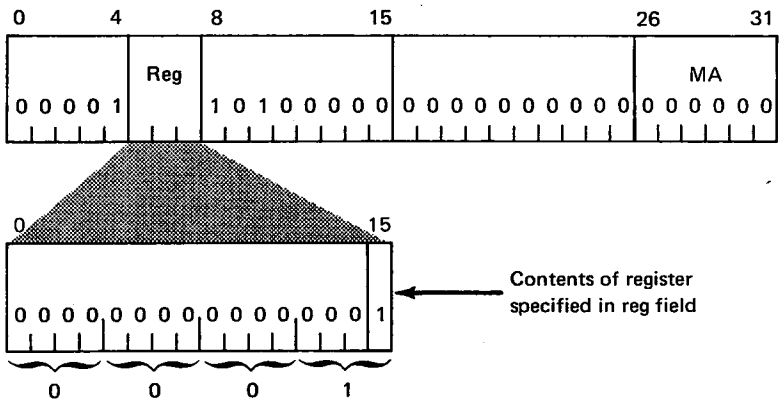
I/O Command

The 5024 uses two operations of the set interrupt command. They are:

1. Initiate I/O.
2. Reset 5024.

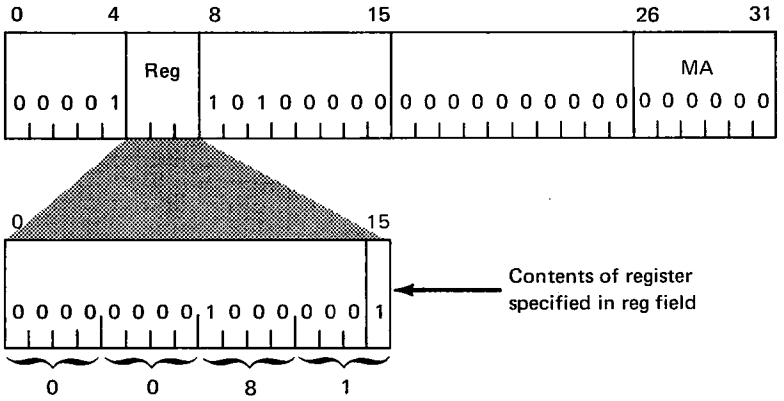
Initiate I/O

Initiate I/O starts the cycle steal of the IOTB by the controller. An interrupt is generated after the cycle steal operation is completed. If the module address (MA) is not zero a program check occurs. The initiate I/O has the following format:



Reset 5024

The reset 5024 resets the printer and reader status words in the controller, stops any cycle steal operations that might be in progress, and conditions the controller to accept commands from the user program. If the module address (MA) is not zero a program check occurs. The instruction has the following format:



Controller Initialization

The 5024 controller must be initialized before a programmed device operation is initiated if one or more of the following conditions have occurred:

- The 5024 power is applied
- After a system reset
- After a Reset 5024 command
- The 5024 is varied off-line and then back on-line.

The 5024 is initialized with information contained in six *data tables.* The information in these data tables will vary with machine configuration and circuit logic level. The tables are provided with each 5024 and are updated when the configuration or circuit logic is changed. The customer engineer will provide the updates to these tables.

It is the user's responsibility to initialize the 5024 using the data tables, or to ensure that a compatible level of MSP/7 5024 support is used.

IOTB Sequence

The initialization is accomplished by a sequence of six IOTB's. One IOTB is defined for each data table. The output diagnostic/initialize controller command, a user defined status and data address, a fixed constant, and a variable word count are used in each IOTB. The constant and word count are obtained from the data tables. Each data table must end with a hex D7FB.

The following is an example of an IOTB initialization seqeunce:

IOTB	Command Word	Status Address	Constant	Word Count	Data Address
Table 1	5000	XXXX	0CE8	0008	AAAA
Table 2	5000	XXXX	0DA0	0030	BBBB
Table 3	5000	XXXX	0F02	000F	CCCC
Table 4	5000	XXXX	0FC0	0011	DDDD
Table 5	5000	XXXX	0FF0	0006	EEEE
Table 6	5000	XXXX	0D9A	0002	FFFF
		User Defined			User Defined

If an improper sequence of IOTB's or incorrect data tables are used the results will be unpredictable.

Error Recovery

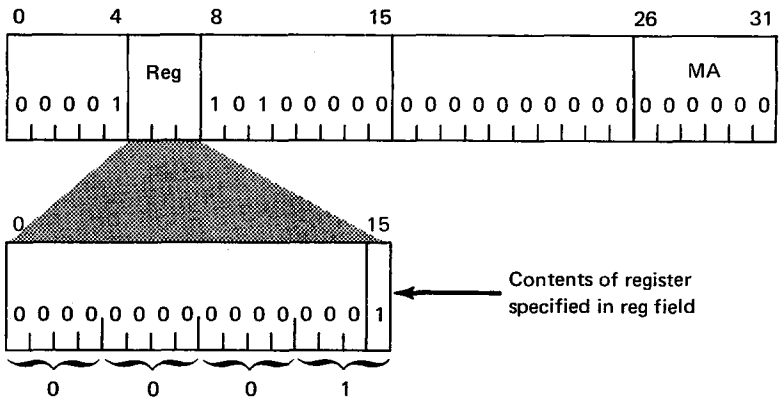
If the output diagnostic/initialize controller command is unsuccessful and the 5024 is on-line to the System/7, a reset 5024 command must be issued and the initialization retried. If retry fails, verify the IOTBs and data tables and retry the operation.

IBM 2502 CARD READER

The 2502 Card Reader accepts input from standard 80 column punched cards, with a maximum throughput of 300 cards per minute. The data controller transfers this input by cycle stealing when the input has been read from one card. The operator can initial program load (IPL) the System/7 from the 2502, except with SBCA (Sensor Based Control Adapter, RPQ D08119).

Initiate I/O

Cycle steal occurs when an initiate I/O is directed to the 2502. This fetches the IOTB which contains the control parameters. If the module address (MA) is not zero a program check occurs. Initiate I/O format is:



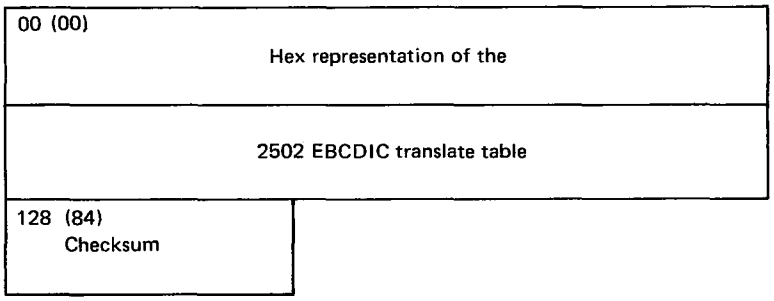
Initial Program Load (IPL)

The 2502 Reader will support operator initiated IPL (except with SBCA) but not auto-restart IPL. Card format is punched card code (Hollerith). Each four card columns will be translated to Hex digits (0-F) and stored in processor storage as a word, starting in location 0000. The first four columns of card 1 will be stored as the first word, in storage location 0000. A total of 1224 columns (17 cards) of punched information is required to IPL 306 words in storage. Cards are read sequentially until all 1224 columns have been read. Columns 73–80 are reserved for ID and/or sequence numbers. These columns are not included in the required 1224 data columns and will not be transmitted to storage.

Characters 0-9 and A-F are translated to their corresponding hex code. A blank card column will translate to a hex 0. Non-valid punches will cause a validity check.

2502 Card Reader Translate Table

A 2502 card reader translate table is required by the 2502 prior to issuing a read and translate card code mode command. The translate table has the following format:



Translate Table

Each 8-bit byte of the translate table is the value that the corresponding card column is translated to and placed in System/7 memory as a result of reading a card, e.g., a card column having punches 12-1 will be converted to a value using a look-up in the table to obtain the hex value of C1. The IBM program module $XL2502 card reader translate table may be referenced to create the translate table.

The description in the table following will provide the punches corresponding to the hex value.

Hexadecimal Value	Card Code	Hexadecimal Value	Card Code
40F1	No punches, 1	D2D3	11-2, 11-3
F2F3	2,3	D4D5	11-4, 11-5
F4F5	4,5	D6D7	11-6, 11-7
F6F7	6,7	D859	11-8, 11-1-8
F8F9	8,1-8	5A5B	11-2-8, 11-3-8
7A7B	2-8,3-8	5C5D	11-4-8, 11-5-8
7C7D	4-8,5-8	5E5F	11-6-8, 11-7-8
7E7F	6-8,7-8	D911	11-9, 11-1-9
F931	9,1-9	1213	11-2-9, 11-3-9
3233	2-9,3-9	1415	11-4-9, 11-5-9
3435	4-9,5-9	1617	11-6-9, 11-7-9
3637	6-9,7-9	1819	11-8-9, 11-1-8-9
3839	8-9,1-8-9	1A1B	11-2-8-9, 11-3-8-9
3A3B	2-8-9,3-8-9	1C1D	11-4-8-9, 11-5-8-9
3C3D	4-8-9,5-8-9	1E1F	11-6-8-9, 11-7-8-9
3E3F	6-8-9,7-8-9	D0A1	11-0, 11-0-1
F061	0,0-1	A2A3	11-0-2, 11-0-3
E2E3	0-2,0-3	A4A5	11-0-4, 11-0-5
E4E5	0-4,0-5	A6A7	11-0-6, 11-0-7
E6E7	0-6,0-7	A8A0	11-0-8, 11-0-1-8
E869	0-8,0-1-8	AAAB	11-0-2-8, 11-0-3-8
E06B	0-2-8, 0-3-8	ACAD	11-0-4-8, 11-0-5-8
6C6D	0-4-8, 0-5-8	AEAF	11-0-6-8, 11-0-7-8
6E6F	0-6-8, 0-7-8	A9E1	11-0-9, 11-0-1-9
E921	0-9, 0-1-9	6263	11-0-2-9, 11-0-3-9
2223	0-2-9, 0-3-9	6465	11-0-4-9, 11-0-5-9
2425	0-4-9, 0-5-9	6667	11-0-6-9, 11-0-7-9
2627	0-6-9, 0-7-9	6820	11-0-8-9, 11-0-1-8-9
2829	0-8-9, 0-1-8-9	EAEB	11-0-2-8-9, 11-0-3-8-9
2A2B	0-2-8-9, 0-3-8-9	ECED	11-0-4-8-9, 11-0-5-8-9
2C2D	0-4-8-9, 0-5-8-9	EEEF	11-0-6-8-9, 11-0-7-8-9
2E2F	0-6-8-9, 0-7-8-9	50C1	12, 12-1
60D1	11,11-1	C2C3	12-2, 12-3

Hexadecimal Value	Card Code
C4C5	12-4, 12-5
C6C7	12-6, 12-7
C849	12-8, 12-1-8
4A4B	12-2-8, 12-3-8
4C4D	12-4-8, 12-5-8
4E4F	12-6-8, 12-7-8
C901	12-9, 12-1-9
0203	12-2-9, 12-3-9
0405	12-4-9, 12-5-9
0607	12-6-9, 12-7-9
0809	12-8-9, 12-1-8-9
0A0B	12-2-8-9, 12-3-8-9
0C0D	12-4-8-9, 12-5-8-9
0E0F	12-6-8-9, 12-7-8-9
C081	12-0, 12-0-1
8283	12-0-2, 12-0-3
8485	12-0-4, 12-0-5
8687	12-0-6, 12-0-7
8880	12-0-8, 12-0-1-8
8A8B	12-0-2-8, 12-0-3-8
8C8D	12-0-4-8, 12-0-5-8
8E8F	12-0-6-8, 12-0-7-8
8941	12-0-9, 12-0-1-9
4243	12-0-2-9, 12-0-3-9
4445	12-0-4-9, 12-0-5-9
4647	12-0-6-9, 12-0-7-9
4800	12-0-8-9, 12-0-1-8-9
CACB	12-0-2-8-9, 12-0-3-8-9
CCCD	12-0-4-8-9, 12-0-5-8-9
CECF	12-0-6-8-9, 12-0-7-8-9
6A91	12-11, 12-11-1
9293	12-11-2, 12-11-3
9495	12-11-4, 12-11-5
9697	12-11-6, 12-11-7
9890	12-11-8, 12-11-1-8
9A9B	12-11-2-3, 12-11-3-3
9C9D	12-11-4-8, 12-11-3-8
9E9F	12-11-6-8, 12-11-7-8
9951	12-11-9, 12-11-1-9
5253	12-11-2-9, 12-11-3-9
5455	12-11-4-9, 12-11-5-9
5657	12-11-6-9, 12-11-7-9
5810	12-11-8-9, 12-11-1-8-9
DADB	12-11-2-8-9, 12-11-3-8-9
DCDD	12-11-4-8-9, 12-11-5-8-9
DEDF	12-11-6-8-9, 12-11-7-8-9
70B1	12-11-9, 12-11-0-1
B2B3	12-11-0-2, 12-11-0-3
B4B5	12-11-0-4, 12-11-0-5
B6B7	12-11-0-6, 12-11-0-7
B8B0	12-11-0-8, 12-11-0-1-8
BABB	12-11-0-2-8, 12-11-0-3-8
BCBD	12-11-0-4-8, 12-11-0-5-8
BEBF	12-11-0-6-8, 12-11-0-7-8
B971	12-11-0-9, 12-11-0-1-9
7273	12-11-0-2-9, 12-11-0-3-9
7475	12-11-0-4-9, 12-11-0-5-9
7677	12-11-0-6-9, 12-11-0-7-9
7830	12-11-0-8-9, 12-11-0-1-8-9
FAFB	12-11-0-2-8-9, 12-11-0-3-8-9
FCFD	12-11-0-4-8-9, 12-11-0-5-8-9
FEFF	12-11-0-6-8-9, 12-11-0-7-8-9

Check Sum

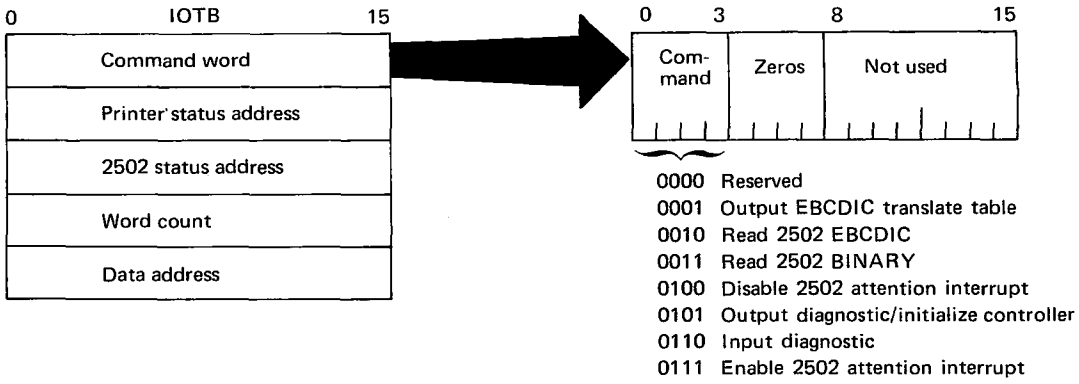
The check sum algorithm for the translate table begins with the hex value of X'FFFF' and consecutively exclusive OR's ($\forall$) the first 128 words of the translate table. This is shown in the following format:

checksum = X'FFFF' $\forall$ W01 $\forall$ W02 $\forall$ W03 • • • $\forall$ W128

I/O Transfer Block

The address of the IOTB must be stored at location 0008 prior to an initiate I/O command.

After the initiate I/O instruction is accepted, the 5024 controller starts to cycle steal the IOTB. The controller will ignore the unused control parameters (printer status address and word count). One full card (80 columns) is read whenever a card read operation is completed. In order to read another card, another initiate I/O must be issued.



Command Word

The command word defines the operation to be performed. Bits 0-3 are the command, bits 4-15 are zeros when used with the 2502. Five commands within the command word are used by the user program.

- *Ouput EBCDIC Translate Table 0001.* This command transfers the EBCDIC translate table from main storage to the card reader attachment, and must be executed without error before a Read 2502 EBCDIC command can be executed. For a description of the translate table see 2502 Card Read Translate Table in this chapter.
- *Read and Translate Card Code Mode 0010.* The 2502 reads the 256 hole patterns of 80-column card code. Translation is from punched card code to EBCDIC. Each 16 bit word transferred to storage contains information from two card columns. If this command is issued before command 0001, bits 1 and 2 will be set on in status word 1 and the command will not be executed.
- *Read Binary Mode 0011.* The 2502 reads column binary, all bits being valid. No translation takes place. Each card column represents a 16 bit word with Hole 12 being the high-order bit (bit 0). The low-order four bits (bits 12-15) are forced to a zero state.
- *Disable Attention Interrupt 0100.* This command will disable the 2502 Attention Interrupt. Any initiate I/O command (other than the disable command) issued to the reader, re-enables the attention interrupt.
- *Input Diagnostic 0110.* This command is used by the customer engineer for diagnostic purposes. If this command is used in a user program, unpredictable results will occur.

- *Output Diagnostic/Initialize Controller 0101.* This command is used to initialize the 5024 controller using data tables. The 2502 attention interrupt must be disabled prior to issuing this command. For further definition of 5024 controller initialization see Controller Initialization in this chapter.
- *Enable Attention Interrupt 0111.* This command will enable the 2502 Attention Interrupt.

2502 Status Address

This word contains a user defined storage address of the status buffer.

Word Count

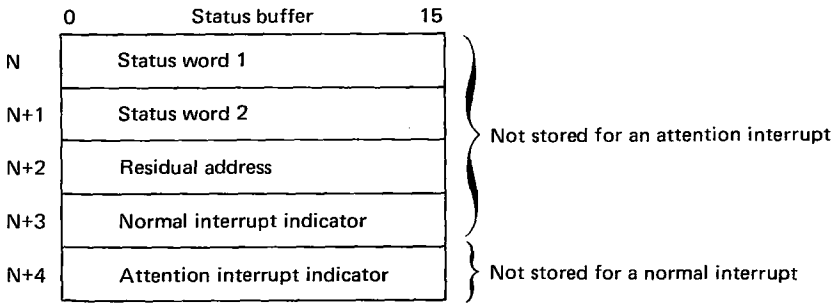
The controller establishes the word count for all the 2502 commands.

Data Address

The data address is the address of the buffer in the System/7 into which data is transferred. This parameter is fetched by the controller which holds it in a register, updates it, and presents it during status transfer.

Status Buffer

Before the 2502 attachment presents an interrupt request, 5 control words are placed in System/7 storage for program interrogation.



N is the address contained in the 2502 status word address of the IOTB.

Status Word 1.

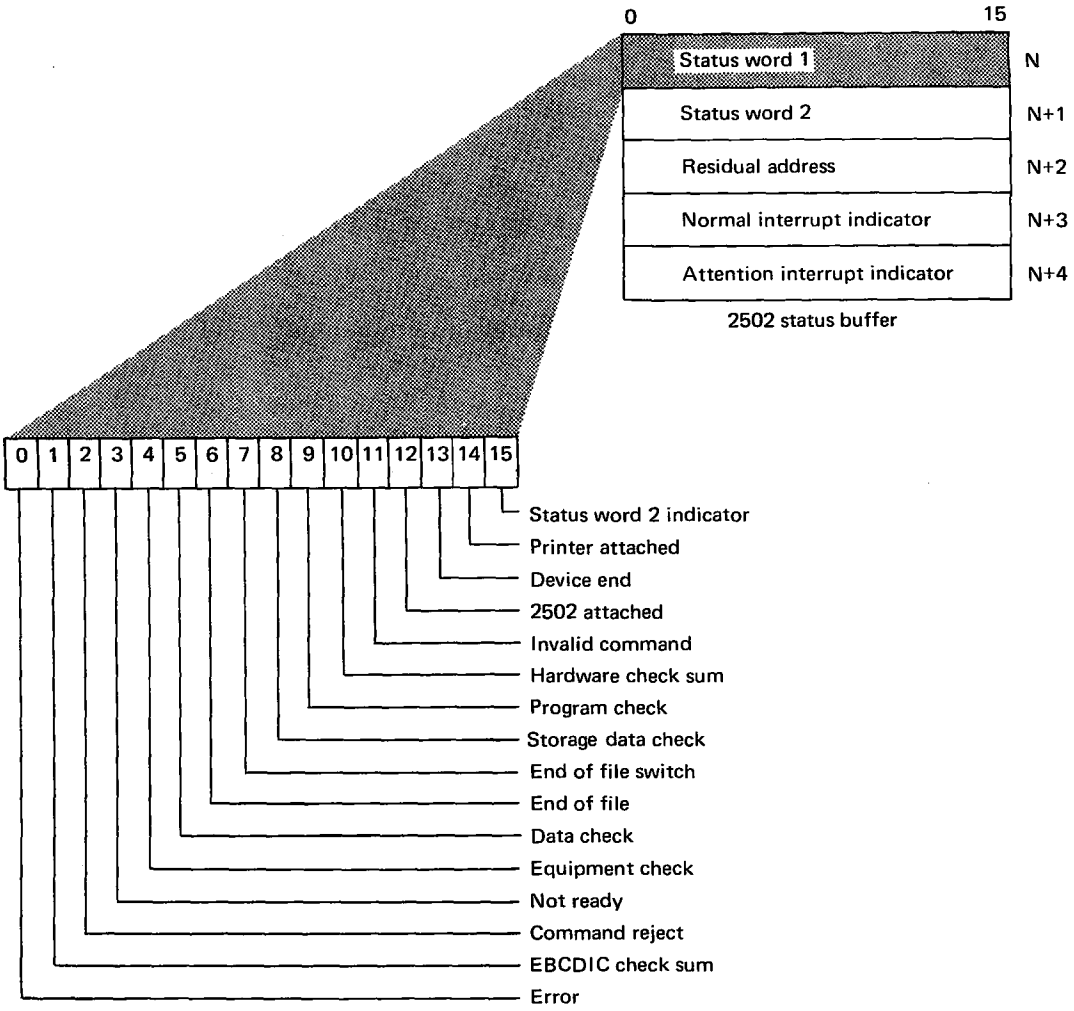


Figure 17-5. 2502 status word 1

Bit	
0	Error/Exception Status Bit—This bit is on whenever any error bit is set or an exception condition occurs.
1	EBCDIC Check Sum—This bit is on when an error is detected in the EBCDIC translate table.
2	Command Reject—This bit is set when the attachment is not capable of executing the command.
3	Reader not Ready.
4	Equipment Check—This bit is set for one of the following: 1. Feed Check 2. Emitter Sync Check 3. Hardware check Further information as to cause is found in status word 2.
5	Data Check—This bit is set for a validity check.
6	End of File—This bit is set when EOF switch is on, the hopper is empty and the last card has been read.
7	End of File Switch—This bit is set on when the EOF switch is in the ON position.
8	Storage Data Check—The storage location accessed by the attachment during the current instruction contained bad parity.
9	Program Check—The main storage address presented by the attachment during the current cycle steal input/output operation for data exceeds storage size installed in the system.
10	Hardware Check Sum Error.
11	Invalid Command—This bit is set any time an invalid 2502 is issued.
12	2502 Attached.
13	Device End—This bit will be set whenever the attachment/device operation is terminated. This bit generates a interrupt request to alert the program that the attachment may accept new commands.
14	Printer Attached.
15	Status Word 2 Indicator—This bit is set to indicate additional information may be found in Status Word 2.

Status Word 2.

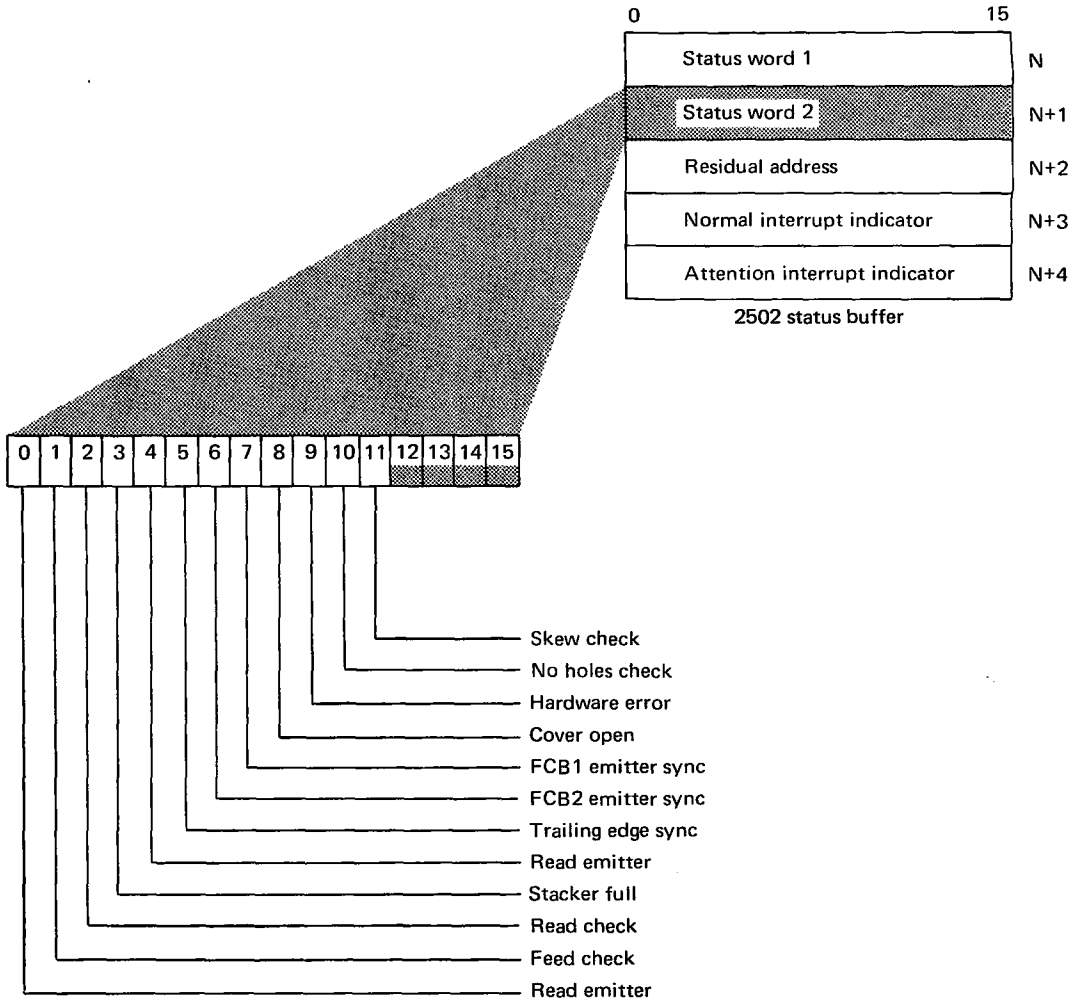


Figure 17-6. 2502 status word 2

Bit	
0	Read Emitter Sync Check (emitter pulse missing)
1	Feed Check
2	Read Check (All holes are not sensed at end of card)
3	Stacker Full or Jam
4	Read Emitter Sync Check (emitter pulse is always present)
5	Trailing Edge Sync Check or Jam
6	FCB2 Emitter Sync Check or Jam*
7	FCB1 Emitter Sync Check or Jam*
8	Cover Open
9	Hardware Error (error is detected in attachment circuit)
10	No Holes Check (bit is on if a hole is detected prior to column 0)
11	Skew check (data is detected at emitter 16 time)
12	Reserved (ZERO)
13	Reserved (ZERO)
14	Reserved (ZERO)
15	Reserved (ZERO)

**Note*. This bit will be set to 1 if a circuit breaker in the 2502 fails to operate properly. If the bit is set to 1 the 2502 may have failed to read properly.

Residual Address. The address of the last data location accessed by the initiate I/O command.

Normal Interrupt Indicator. Bit 0 is set to 1 after the card has been read and an interrupt is about to be sent to the System/7.

Attention Interrupt Indicator. Bit 0 is set to 1 as a result of the 2502 going ready from the not ready state.

Interrupt Presentation

The 2502 Attachment can interrupt on level 3 sublevel 1 or level 1 sublevel 1. The level and displacement are controlled by wiring chosen at initial installation and cannot be altered by programming. When the processor services the interrupt request, the 2502 Attachment presents a device address of 011100.

Device Not Ready. The read commands result in an interrupt presentation with bits 2 and 3 on in Status Word 1.

Not Attached/Offline. When the 2502 attachment is under control of the CE Panel or power to the 5024 is off, code 3 is returned in response to an initiate I/O command.

Operating and Program Restrictions

Cards are placed in the hopper, 9 edge first and face down. Pressing the start key on the 2502 signals the control unit that the 2502 is ready to feed cards. The control unit initiates a feed cycle, feeding a card from the hopper to the preread station. The control unit makes the card reader ready and indicates this condition by sending an attention interrupt to the System/7, if attention interrupt is enabled.

Timing Information

Throughput speed is 300 cards per minute. System decision time to maintain this throughput is 50ms. Slower speeds may result dependent on the rate that I/O commands are issued by the using program.

If a read request is not received in 30 seconds, the attachment will stop the motor (time out). The reader will remain ready, and the attachment will start the motor again when a read request is received. A time delay of 600ms is required to bring the motor up to speed.

Power Transitions and Resets

If the 2502 is in the ready condition when a power interruption occurs, the ready condition is dropped and manual intervention is required to restore the ready state.

Switches and Indicators

The operators panel on the 2502 contains the switches and light indicators with the exception of the power switch which is located on the 5024. Figure 17-7 illustrates the 2502 operators panel.

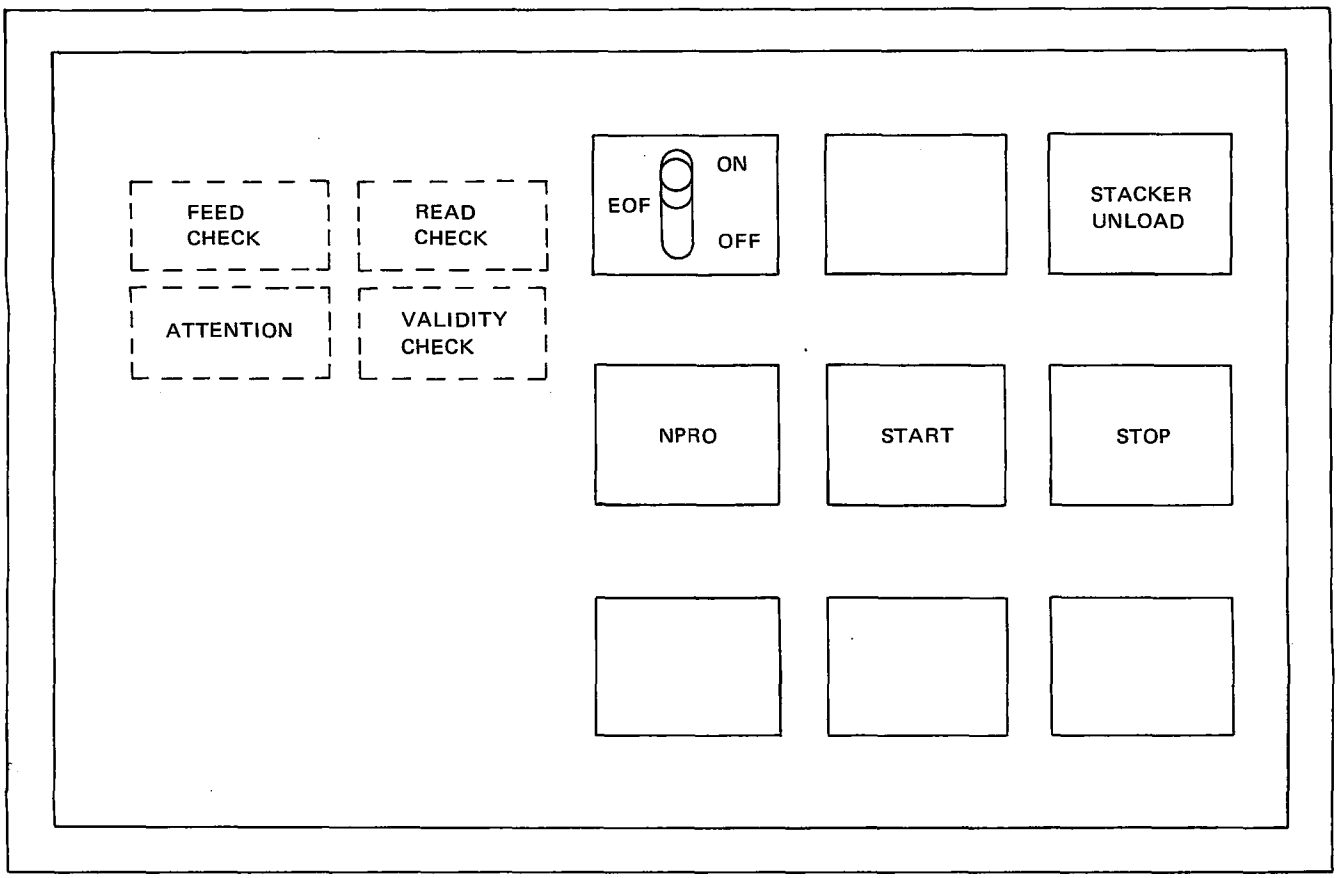


Figure 17-7. 2502 operators panel

Power Switch

The power to the 2502 is controlled by an ON/OFF switch located on the 5024.

Stop Switch

Pressing the stop switch signals the 2502 to stop the transport at the end of the cycle in process.

Start Switch

Pressing the start switch signals the attachment to place the reader in a ready condition.

NPRO Switch

Pressing the NPRO switch signals the attachment to do a non-process runout (NPRO) of any cards in the feed path.

Stacker Unload

Pressing this key stops card feeding to allow the operator approximately 20 seconds to unload cards from the stacker. Pressing the start switch will override this timeout.

End of File

Placing this switch in the ON position conditions the system to end the job in process after the last card has been read from the hopper.

Read Check Light

The read check light indicates that read check bit 2 of status word 2 is on or that a card jam has occurred. When the error condition is corrected, pressing the NPRO switch turns off the light.

Feed Check Light

The feed-check light indicates that either the hopper misfed or a preread station jam has occurred.

Attention Light

The attention light indicates that operator intervention is required to correct one of the following conditions:

- Stacker full
- Cover interlock open
- Stacker jam
- Hopper empty
- Transport cover switch open

The attention light may be turned on by a power-on-reset.

Validity Check Light

The validity check light indicates that the system detected an invalid card code.

Error Checks

Error conditions detected by the attachment will stop the transport and turn on the appropriate light.

The printer operates at 6 lines per inch at 40, 80, 120, or 155 lines per minute depending on character set lengths of 128, 96, 64, or 48 characters respectively. There are 132 print positions. Printing is accomplished by 66 rear double-wide hammers which selectively force the paper against an inked ribbon and print belt. The print belt is a 48-inch continuous steel belt with 192 etched characters.

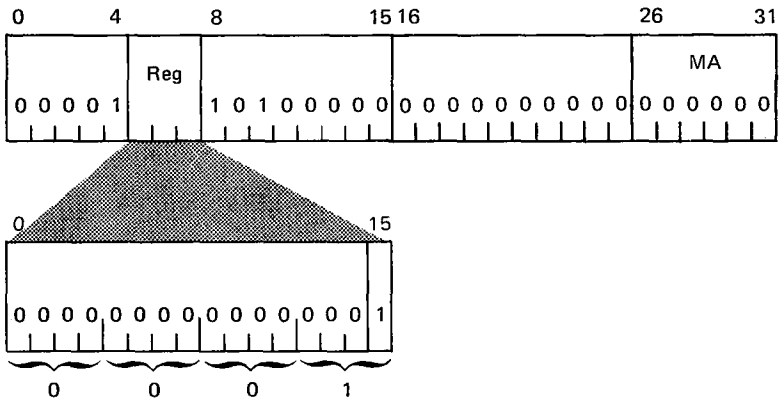
The pin feed carriage handles up to six part forms. Use of continuous card stock forms is not permited. A single line space takes 34ms and the carriage skip speed is 12 inches per second. The printer is controlled by a cycle stealing attachment which contains a print line buffer and a belt image buffer. Printing and carriage movement are accomplished through the use of common logic controls and a print hammer and control the buffer/driver. The attachment is code independent and can accept EBCDIC, ASCII, or any other 8 bit code.

The print belt is operator interchangeable and the adapter will accept all standard 5024 print belts. The user must supply the attachment with the proper belt image.

The carriage is controlled by an electronic Vertical Forms Control (VFC). The user program supplies forms length and overflow line information. Once the forms are aligned by the operator, the carriage can skip to any line on the form or space up to 123 lines.

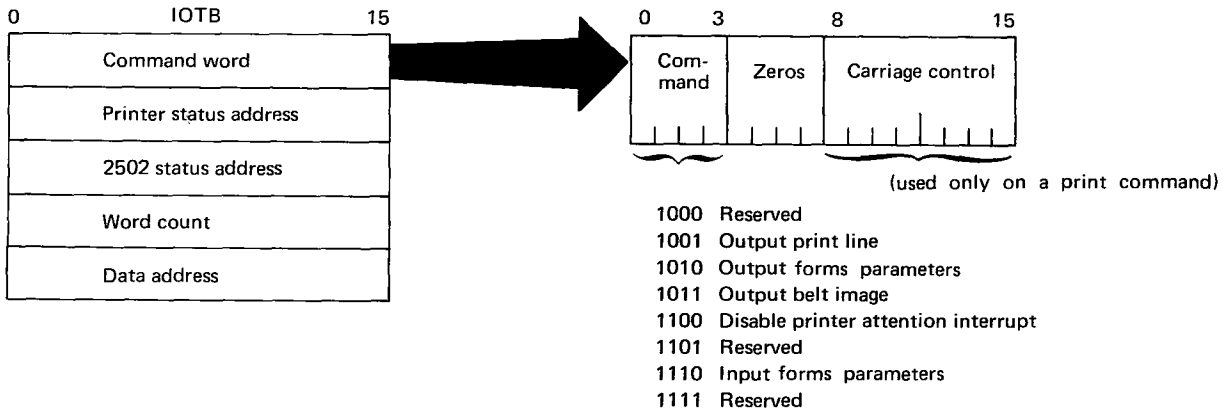
Initiate I/O

The printer attachment is programmed via the Initiate I/O command. When an Initiate I/O command is issued, the printer attachment cycle steals the command word from the IOTB to determine what data will be transferred from main storage to the printer. If the module address (MA) is not zero a program check occurs. The initiate I/O has the following format:



I/O Transfer Block

The IOTB address must be stored at location 0008 before an initiate I/O is issued. After the initiate I/O instruction is accepted, the controller cycle steals the IOTB.



Command Word

The command word defines the operation to be performed. Bits 0-3 identify the command. Bits 4-7 are zero and bits 8-15 are used for carriage and control information on a print command. The commands are as follows:

- *Output Print 1001*. When the print command is detected, the low order byte of the command word contains the carriage control byte.

  If the carriage control byte has a value from 1 to 132 (Hex 84), the carriage will skip *to* the line indicated by the value.

  If the carriage control byte has a value greater than Hex 84, the controller will treat the value as a negative number and cause a space *of* the twos-complement of the binary value.

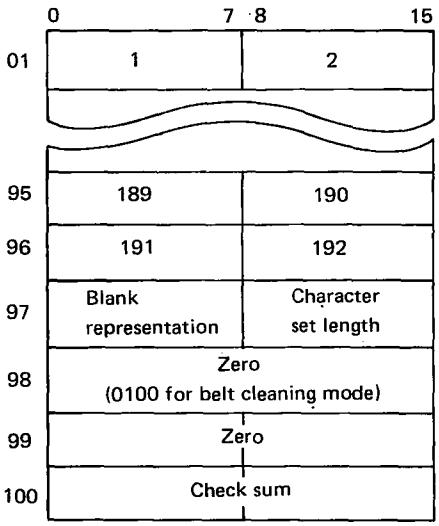
  Therefore, a value of Hex 85 will cause spacing over 123 lines, and a value of Hex FF will cause a one-line space. A value of Hex 00 causes no carriage motion. The line printer performs the carriage space or skip prior to printing the line. Up to 66 words of print data may be transferred for one print line.

  The ranges of skip/space values are shown as follows:

Hex	Binary	Decimal
01	0000 0001	Skip to line 1
	to	
84	1000 0100	Skip to line 132
85	1000 0101	Space 123 lines
	to	
FF	1111 1111	Space 1 line
00	0000 0000	No carriage motion

- *Output Forms Parameters 1010*. This command initiates a one-word cycle steal to the printer attachment. The high order byte is the binary number of lines on the form. The low order byte is the binary value at which the form overflow bit in status word 1 is turned on. The word count is assumed to be 1 and is not checked.

- *Output Belt Image 1011.* This command starts the transfer of belt image data from main storage to the printer attachment. The belt image data field includes the representation of the printable characters in the order they appear on the print belt. The belt image is 100 words and the following format:



– *Words 01 thru 96.* These locations contain the 192 positions of the belt image:

a. 48 character set contained 4 times.
b. 64 character set contained 3 times.
c. 96 character set contained 2 times.
d. 128 character set contained 1 and 1/2 times.

Note. Refer to Figure 17-8 for the relative belt positions, characters, and EBCDIC reference numbers for the four character sets.

– *Blank Representation.* The representation of the character used to print a blank character.
– *Character Set Length.* A value, of 47, 63, 95, or 127 normally is in the low order byte. This represents the number of characters detected between home pulses. This count is unpredictable after power on reset until the print belt is brought up to speed and into sync.

Characters on belt	Value in character set length field	
	Hex	Binary
48	2F	00101111
64	3F	00111111
96	5F	01011111
128	7F	01111111

– *Word 98.* If this word contains a hex value of 0100, a 128 character set mode is forced for belt cleaning purposes.
– *Check Sum.* The checksum algorithm for the belt image begins with the hex value of X'FFFF' and consecutively exclusive OR's (∀) the first 99 words of the belt image. This is shown in the following format: Checksum = X'FFFF'∀W01∀W02∀W03....∀W99

	48 Character Set		64 Character Set		96 Character Set		128 Character Set		
Belt Position	Character	EBCDIC Reference Number (Hex)	Character	EBCDIC Reference Number (Hex)	Character	EBCDIC Reference Number (Hex)	Character		EBCDIC Reference Number (Hex)
1	1	F1	1	F1	1	F1	0		F0
2	2	F2	2	F2	2	F2	1		F1
3	3	F3	3	F3	3	F3	2		F2
4	4	F4	4	F4	4	F4	3		F3
5	5	F5	5	F5	5	F5	4		F4
6	6	F6	6	F6	6	F6	5		F5
7	7	F7	7	F7	7	F7	6		F6
8	8	F8	8	F8	8	F8	7		F7
9	9	F9	9	F9	9	F9	8		F8
10	0	F0	0	F0	0	F0	9		F9
11	=	7E	#	7B	#	7B	−		60
12	)	5D	@	7C	@	7C	、		6B
13	/	61	/	61	/	61	。		4B
14	S	E2	S	E2	S	E2	*		5C
15	T	E3	T	E3	T	E3	゛	Kana Sona	BE
16	U	E4	U	E4	U	E4	ア	A	81
17	V	E5	V	E5	V	E5	イ	I	82
18	W	E6	W	E6	W	E6	ウ	U	83
19	X	E7	X	E7	X	E7	エ	E	84
20	Y	E8	Y	E8	Y	E8	オ	O	85
21	Z	E9	Z	E9	Z	E9	カ	Ka	86
22	&	50	¤	90	¤	90	キ	Ki	87
23	'	6B	'	6B	'	6B	ク	Ku	88
24	(	4D	%	6C	%	6C	ケ	Ke	89
25	J	D1	J	D1	J	D1	コ	Ko	8A
26	K	D2	K	D2	K	D2	サ	Sa	8C
27	L	D3	L	D3	L	D3	シ	Shi	8D
28	M	D4	M	D4	M	D4	ス	Su	8E
29	N	D5	N	D5	N	D5	セ	Se	8F
30	O	D6	O	D6	O	D6	ソ	So	90
31	P	D7	P	D7	P	D7	タ	Ta	91
32	Q	D8	Q	D8	Q	D8	チ	Chi	92
33	R	D9	R	D9	R	D9	ツ	Tsu	93
34	−	60	−	60	−	60	テ	Te	94
35	$	5B	$	5B	$	58	ト	To	95
36	*	5C	*	5C	*	5C	ナ	Na	96
37	A	C1	A	C1	A	C1	ニ	Ni	97
38	B	C2	B	C2	B	C2	ヌ	Nu	98
39	C	C3	C	C3	C	C3	ネ	Ne	99
40	D	C4	D	C4	D	C4	ノ	No	9A
41	E	C5	E	C5	E	C5	ハ	Ha	9D
42	F	C6	F	C6	F	C6	ヒ	Hi	9E
43	G	C7	G	C7	G	C7	フ	Pu	9F
44	H	C8	H	C8	H	C8	ヘ	He	A2
45	I	C9	I	C9	I	C9	ホ	Ho	A3
46	+	4E	+	4E	+	4E	マ	Ma	A4
47	.	4B	.	4B	.	4B	ミ	Mi	A5
48	'	7D	⊕	FF	⊕	FF	ム	Mu	A6
49	1	F1	(	4D	(	4D	メ	Me	A7
50	2	F2	<	4C	<	4C	モ	Mo	A8
51	3	F3	[	4A	[	4A	ヤ	Ya	A9
52	4	F4	!	4F	!	4F	ユ	Yu	AA
53	5	F5	)	5D	)	5D	ヨ	Yo	AC
54	6	F6	]	5A	]	5A	ラ	Ra	AD
55	7	F7	;	5E	;	5E	リ	Ri	AE
56	8	F8	¬	5F	¬	5F	ル	Ru	AF
57	9	F9	\	E0	\	E0	レ	Re	BA
58	0	F0	−	6D	−	6D	ロ	Ro	BB
59	=	7E	>	6E	>	6E	ワ	Wa	BC
60	)	5D	?	6F	?	6F	ヲ	Wo	46
61	/	61	:	7A	:	7A	ン	Un	BD
62	S	E2	=	7E	=	7E	ッ	Sm Tsu	56
63	T	E3	"	7F	"	7F	¥	Yen	5B
64	U	E4	'	7D	'	7D	ｐ	a	47

Figure 17-8 (Part 1 of 3). Relative Belt Positions/Characters/EBCDIC

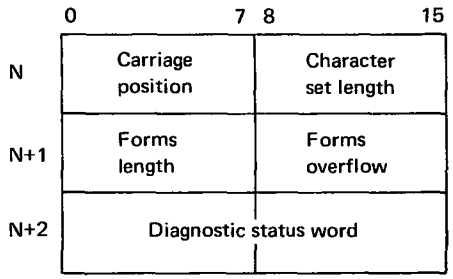
	48 Character Set		64 Character Set		96 Character Set		128 Character Set	
Belt Position	Character	EBCDIC Reference Number (Hex)	Character	EBCDIC Reference Number (Hex)	Character	EBCDIC Reference Number (Hex)	Character	EBCDIC Reference Number (Hex)
65	V	E5	1	F1	:	6A	؛ i	48
66	W	E6	2	F2	a	81	؟ u	49
67	X	E7	3	F3	b	82	ء e	51
68	Y	E8	4	F4	c	83	آ o	52
69	Z	E9	5	F5	d	84	ا Sm. Ya	53
70	&	50	6	F6	e	85	ب Sm. Yu	54
71	,	6B	7	F7	f	86	ة Sm. Yo	55
72	(	4D	8	F8	g	87	ت Sem Sona	BF
73	J	D1	9	F9	h	88	/	61
74	K	D2	0	F0	i	89	A	C1
75	L	D3	#	7B	{	8B	B	C2
76	M	D4	@	7C	}	9B	C	C3
77	N	D5	/	61	~	A1	D	C4
78	O	D6	S	E2	j	91	E	C5
79	P	D7	T	E3	k	92	F	C6
80	Q	D8	U	E4	l	93	G	C7
81	R	D9	V	E5	m	94	H	C8
82	–	60	W	E6	n	95	I	C9
83	$	5B	X	E7	o	96	J	D1
84	*	5C	Y	E8	p	97	K	D2
85	A	C1	Z	E9	q	98	L	D3
86	B	C2	¤	90	r	99	M	D4
87	C	C3	,	6B	&	50	N	D5
88	D	C4	%	6C	x	79	O	D6
89	E	C5	J	D1	s	A2	P	D7
90	F	C6	K	D2	t	A3	Q	D8
91	G	C7	L	D3	u	A4	R	D9
92	H	C8	M	D4	v	A5	S	E2
93	¦	C9	N	D5	w	A6	T	E3
94	+	4E	O	D6	x	A7	U	E4
95	.	4B	P	D7	y	A8	V	E5
96	'	7D	Q	D8	z	A9	W	E6
97	1	F1	R	D9	1	F1	X	E7
98	2	F2	–	60	2	F2	Y	E8
99	3	F3	$	5B	3	F3	Z	E9
100	4	F4	*	5C	4	F4	–	58
101	5	F5	A	C1	5	F5	#	7B
102	6	F6	B	C2	6	F6	$	E0
103	7	F7	C	C3	7	F7	?	6F
104	8	F8	D	C4	8	F8	%	6C
105	9	F9	E	C5	9	F9	@	7C
106	0	F0	F	C6	0	F0	=	7E
107	=	7E	G	C7	#	7B	+	4E
108	)	5D	H	C8	@	7C	£	DB
109	/	61	I	C9	/	61	I	5A
110	S	E2	+	4E	$	E2	<	4C
111	T	E3	.	4B	T	E3	>	6E
112	U	E4	⊕	FF	U	E4	=	6D
113	V	E5	(	4D	V	E5	¯	A1
114	W	E6	<	4C	W	E6	"	7F
115	X	E7	[	4A	X	E7	¯	5F
116	Y	E8	¦	4F	Y	E8	'	7D
117	Z	E9	)	5D	Z	E9	¦	4F
118	&	50	]	5A	¤	90	&	50
119	,	6B	;	5E	,	6B	:	7A
120	(	4D	¯	5F	%	6C	;	5E
121	J	D1	\	E0	J	D1	°	41
122	K	D2	–	6D	K	D2	\	44
123	L	D3	>	6E	L	D3	.	45
124	M	D4	?	6F	M	D4	⌜	42
125	N	D5	:	7A	N	D5	⌟	43
126	O	D6	=	7E	O	D6	(	4D
127	P	D7	"	7F	P	D7	)	5D
128	Q	D8	'	7D	Q	D8	⊕	FF

Figure 17-8 (Part 2 of 3). Relative Belt Positions/Characters/EBCDIC

	48 Character Set		64 Character Set		96 Character Set		128 Character Set	
Belt Position	Character	EBCDIC Reference Number (Hex)	Character	EBCDIC Reference Number (Hex)	Character	EBCDIC Reference Number (Hex)	Character	EBCDIC Reference Number (Hex)
129	R	D9	1	F1	B	D9	0	F0
130	–	60	2	F2	–	60	1	F1
131	$	5B	3	F3	$	5B	2	F2
132	*	5C	4	F4	*	5C	3	F3
133	A	C1	5	F5	A	C1	4	F4
134	B	C2	6	F6	B	C2	5	F5
135	C	C3	7	F7	C	C3	6	F6
136	D	C4	8	F8	D	C4	7	F7
137	E	C5	9	F9	E	C5	8	F8
138	F	C6	0	F0	F	C6	9	F9
139	G	C7	#	7B	G	C7	–	60
140	H	C8	@	7C	H	C8	,	6B
141	I	C9	/	61	I	C9	.	4B
142	+	4E	S	E2	+	4E	*	5C
143	.	4B	T	E3	.	4B	゛	Kana Sona BE
144	'	7D	U	E4	⊕	FF	ア	A 81
145	1	F1	V	E5	(	4D	イ	I 82
146	2	F2	W	E6	<	4C	ウ	U 83
147	3	F3	X	E7	[	4A	エ	E 84
148	4	F4	Y	E8	|	4F	オ	O 85
149	5	F5	Z	E9	)	5D	カ	Ka 86
150	6	F6	¤	90	]	5A	キ	Ki 87
151	7	F7	,	6B	;	5E	フ	Ku 88
152	8	F8	%	6C	¬	5F	ケ	Ke 89
153	9	F9	J	D1	\	E0	コ	Ko 8A
154	0	F0	K	D2	–	6D	サ	Sa 8C
155	=	7E	L	D3	>	6E	シ	Shi 8D
156	)	5D	M	D4	?	6F	ス	Su 8E
157	/	61	N	D5	:	7A	セ	Se 8F
158	S	E2	O	D6	=	7E	ソ	So 90
159	T	E3	P	D7	"	7F	タ	Ta 91
160	U	E4	Q	D8	'	7D	チ	Chi 92
161	V	E5	R	D9	:	6A	ツ	Tsu 93
162	W	E6	–	60	a	81	テ	Te 94
163	X	E7	$	5B	b	82	ト	To 95
164	Y	E8	*	5C	c	83	ナ	Na 96
165	Z	E9	A	C1	d	84	ニ	Ni 97
166	&	50	B	C2	e	85	ヌ	Nu 98
167	,	6B	C	C3	f	86	ネ	Ne 99
168	(	4D	D	C4	g	87	ノ	No 9A
169	J	D1	E	C5	h	88	ハ	Ha 9D
170	K	D2	F	C6	i	89	ヒ	Hi 9E
171	L	D3	G	C7	{	8B	フ	Pu 9F
172	M	D4	H	C8	}	9B	ヘ	He A2
173	N	D5	I	C9	~	A1	ホ	Ho A3
174	O	D6	+	4E	j	91	マ	Ma A4
175	P	D7	.	4B	k	92	ミ	Mi A5
176	Q	D8	⊕	FF	l	93	ム	Mu A6
177	R	D9	(	4D	m	94	メ	Me A7
178	–	60	<	4C	n	95	モ	Mo A8
179	$	5B	[	4A	o	96	ヤ	Ya A9
180	*	5C	|	4F	p	97	ユ	Yu AA
181	A	C1	)	5D	q	98	ヨ	Yo AC
182	B	C2	]	5A	r	99	ラ	Ra AD
183	C	C3	;	5E	&	50	リ	Ri AE
184	D	C4	¬	5F	x'	79	ル	Ru AF
185	E	C5	\	E0	s	A2	レ	Re BA
186	F	C6	–	6D	t	A3	ロ	Ro BB
187	G	C7	>	6E	u	A4	ワ	Wa BC
188	H	C8	?	6F	v	A5	ヲ	Wo 46
189	I	C9	:	7A	w	A6	ン	Un BD
190	+	4E	=	7E	x	A7	ッ	Sm. Tsu 56
191	.	4B	"	7F	y	A8	¥	Yen 5B
192	'	7D	'	7D	z	A9	゜	a 47

Figure 17-8 (Part 3 of 3). Relative Belt Positions/Characters/EBCDIC

- *Disable Attention Interrupt 1100.* This command disables the attention interrupt. Any initiate I/O command (other than the disable command), when issued to the printer, re-enables the attention interrupt.
- *Input Forms Parameter 1110.* The forms parameter modifier causes the printer attachment to cycle steal three words into main storage with the following format:



Word count is assumed to be 3 and is not checked.

– *Carriage Position.* A value from 1 to 132 is in the high order byte. This value is in binary and will present the position of the forms.
– *Character Set Length.* Refer to the description of character set length for the *Output Belt Image 1011* command. To determine set value of the character set length, load any valid belt image buffer, do a print with word count of 0, and no forms motion. Then bring the belt up to speed, approximately 3 seconds, and issue an input forms parameter command.
– *Forms Length.* The high-order byte transferred to the processor contains the forms length. The form length can be altered by system program via a start output forms parameter, and must be issued after each power on reset or when the 5024 is taken off-line and put back on-line.
– *Forms Overflow.* The low order byte is the binary value at which the form overflow bit in status word 1 is turned on. This byte is initialized by the output forms parameter command, and must be issued after each power on reset or when the 5024 is taken off-line and put back on-line.
– *Diagnostic Status Word.* This word contains information for diagnostic testing. These bits are as defined below:

Bit 0 - Sync Check–Installed and supplied character set length do not compare (programming or hardware error).
Bit 1 - Sync Check–Not odd or even subscan 4 at homepulse.
Bit 2 - Sync Check–Not phase 1 on even subscan 4 at homepulse.
Bit 3 - Sync Check–Not phase 4 on odd subscan 4 at homepulse.
Bit 4 - Sync Check–Character count at last homepulse greater than 256.
} Hardware error
Bit 5 - Over Temperature
Bit 6 - Hammer Check
Bit 7 - Not Used (ZERO)
Bit 8 - End of Forms on Manual Space
Bit 9 - End of Forms
Bit 10 - Forms Jam
Bit 11–15 - Not Used (ZERO)

Printer Status Address

This address contains the starting address of the status buffer (5 control words) that is presented on every normal interrupt.

Word Count

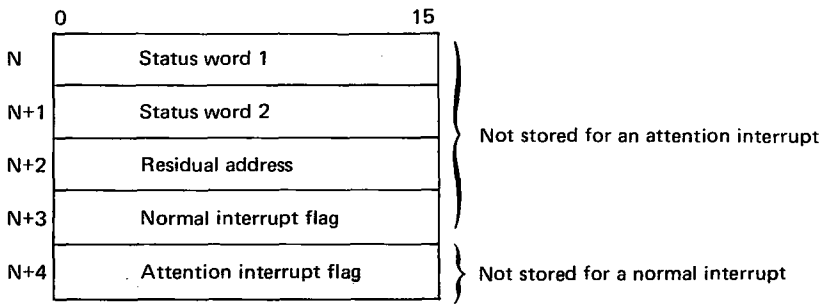
The word count is the number of words to be transferred during the operation. The output print line command requires the user to set a valid word count. The word count will be 0 for a space or skip only operation. If the word count exceeds 66 the data transfer is truncated at 66. The incorrect record length bit is set in status word 1 and the 66 words that were transferred will print. The controller establishes the word count for all other printer commands.

Data Address

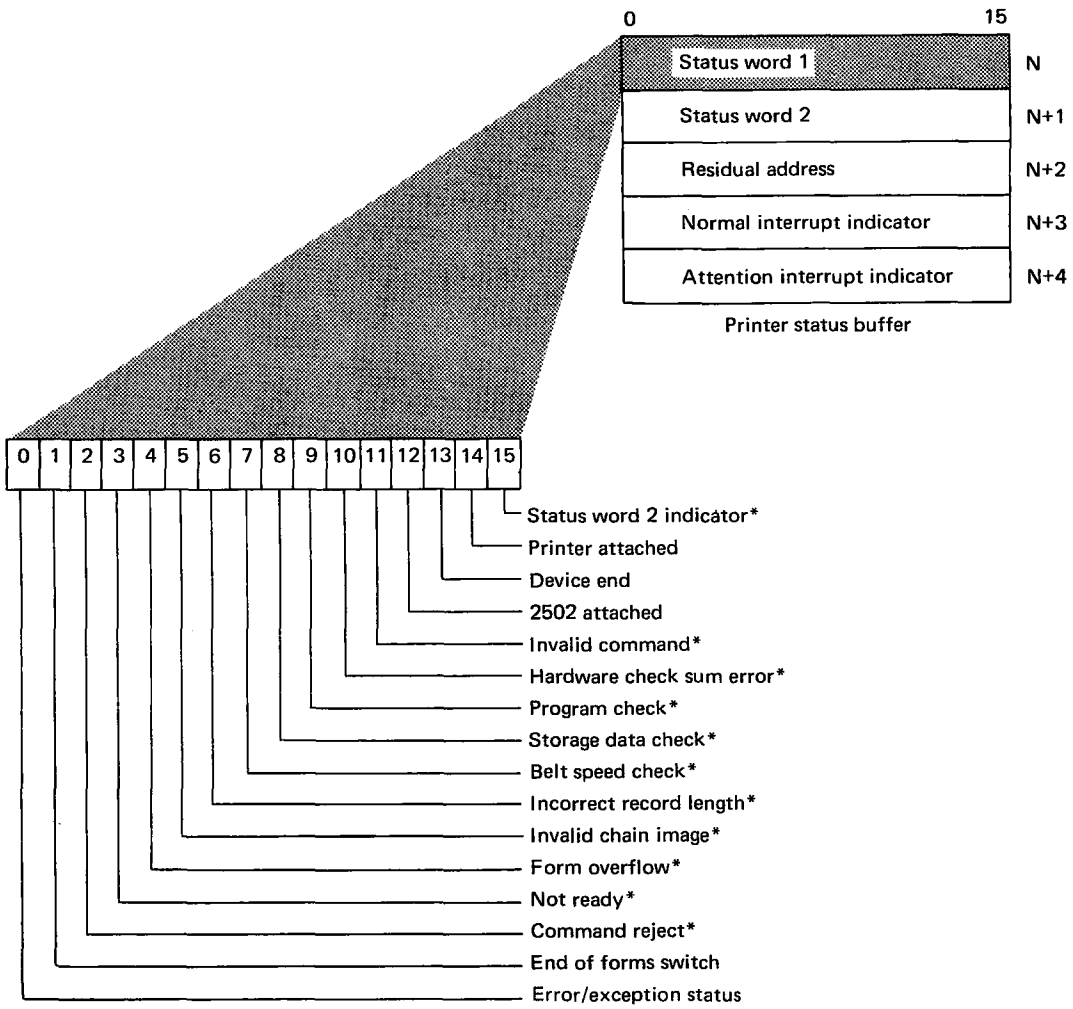
The data address is the address of the buffer in the System/7 to and from which data will be transferred. This parameter is fetched by the controller which holds it in a register, updates it, and presents it to the 5024 during status transfer.

Status Buffer

When the printer attachment presents an interrupt request as a result of an issued command, 5 control words are placed in System/7 storage for program interrogation.



Status Word 1.

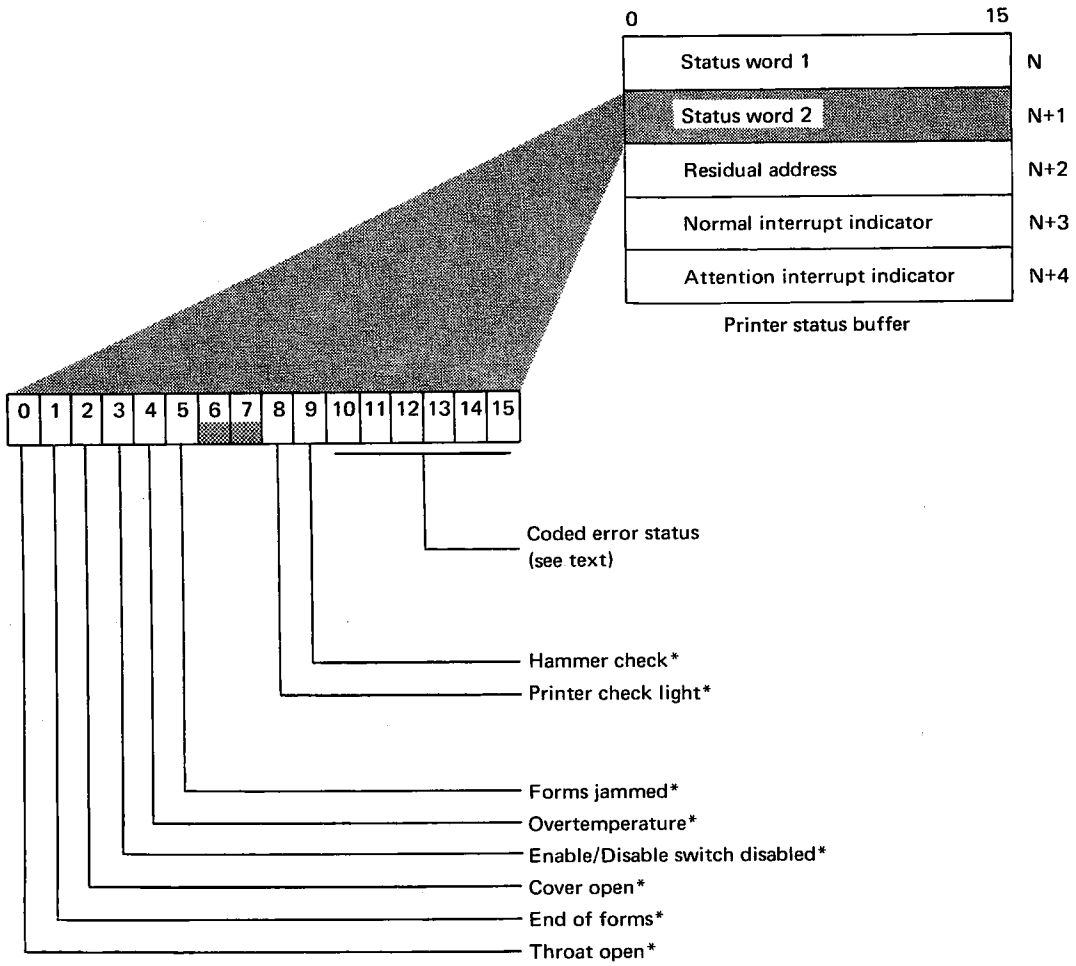


*The error status bit is turned on.

Figure 17-9. Printer status word 1

Bit	
0	Error/Exception Status Bit—An error or exception condition may have occurred during the execution of the command or printer device check may exist.
1	End of Forms Switch—This bit is on when the switch transfers. Bit 0 is not set and printing continues for another 39 lines (approximately) before ready is dropped.
2	Command Reject is set on under the following conditions: a. An input forms parameter command was issued and a 6 bit coded error status was set. b. A 1000, 1101, or 1111 command was issued. c. A belt speed check occurred during a print command. d. A print command was issued that would have caused the printer to print past the carriage overflow line. e. A print command was issued to a not ready printer. f. Over temperature detected when turning on the print belt motor. g. Belt image checksum error detected when bringing print belt up to speed. h. Forms error occurred during print command. i. Program check or invalid storage address when cycle stealing data for a print or diagnostic command. j. A diagnostic command issued with an invalid count or controller address.
3	Not Ready is set on under the following conditions: a. ENABLE/DISABLE switch in DISABLE position b. Cover open c. Throat open d. End of forms e. Forms jam f. Hammer check g. Over temperature
4	Form Overflow—The number of lines printed on the form equals or exceeds the number presented in output forms parameters instruction. This indication will only occur once per form. This bit will never set if you do not define forms length and overflow value after power on reset.
5	Invalid Belt Image—A check sum error was detected on the belt image.
6	Incorrect Record Length—Set when the word count on a print command exceeds 66 words.
7	Belt Speed Check—The belt did not come up to speed after having been on for two seconds.
8	Storage Data Check—A storage location accessed by an attachment during the current instruction contained bad parity.
9	Program Check—The main storage address presented by the attachment during the current cycle steal operation exceeds the storage size installed in the system.
10	Hardware Check Sum Error—An error has been found on the attachment card.
11	Invalid Command—A command other than those listed under initiate I/O was issued to the printer attachment.
12	2502 Attached.
13	Device End—Device End will be on at the end of an interrupt-producing command.
14	Printer Attached.
15	Status Word 2 Indicator—Indicates printer device information is present in Status Word 2 and should be interrogated.

Status Word 2.



*A coded error status is set.

Figure 17-10. Printer status word 2

Bit	
0	Throat Open–The throat of the printer is open.
1	End of Forms–Printer is out of forms.
2	Cover Open–The cover of the printer is open.
3	Enable/Disable Switch–Switch is in disable position.
4	Over Temperature–The printer has detected an over temperature condition. When this condition occurs, the 24 volt printer power drops and the other status bits are no longer valid.
5	Forms Jammed–The printer was issued a skip or space command and no paper motion was detected.
6	Zero
7	Zero
8	Printer Check Light–Read forms parameter and examine bits 0–7 of the diagnostic status word to determine cause of error condition.
9	Hammer Check–The attachment detected a hammer on when not expected. When this occurs, the 24 volt printer power drops and the other status bits are no longer valid.
10–15	Coded Error Status–These bits are binary coded error information as follows:

Error Code	*Detected Attachment Error*
00	No error
01	Paper clamp (OFF-should be ON)
02	Paper clamp (ON-should be OFF)
03	Power on reset (ON-should be OFF)
04	24V contactor (ON-should be OFF)
05	(Not used)
06	Printer check light (ON-should be OFF)
07	Printer check light (OFF-should be ON)
08	Forms check light (ON-should be OFF)
09	Forms check light (OFF-should be ON)
0A	Carriage step attention (ON-should be OFF)
0B	Carriage last step (OFF-should be ON)
0C	Carriage go (ON-should be OFF)
0D	Carriage go (OFF-should be ON)
0E	Belt go (ON-should be OFF)
0F	Belt go (OFF-should be ON)
10	Ready indicator (OFF-should be ON)
11	Ready indicator (ON-should be OFF)
12	(Not used)
13	(Not used)
14	(Not used)
15	(Not used)
16	(Not used)
17	(Not used)
18	(Not used)
19	Printer overtemperature
1A	Ready indicator powered (OFF-should be ON)
1B	Printer check light powered (ON-should be OFF)
1C	Printer check light powered (OFF-should be ON)
1D	Forms check light powered (ON-should be OFF)
1E	Forms check light powered (OFF-should be ON)
1F	Ready indicator powered (ON-should be OFF)

The occurrence of any of the above errors should be considered a hardware error.

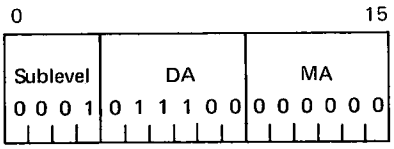
Residual Address. This word contains the address of the last data location accessed by the initiate I/O command.

Normal Interrupt Flag. Bit 0 is a 1 on all interrupts from an initiate I/O command.

Attention Interrupt Flag. As a result of the printer going ready after being in the not ready state, bit 0 is set to a 1 and an interrupt is generated. An initiate I/O command must be addressed to the printer attachment following any reset before the attachment can generate an attention interrupt.

Interrupt Presentation

The printer attachment can interrupt on level 3 sublevel 1 or level 1, sublevel 1. The level and displacement is controlled by wiring and cannot be altered by programming. When the processor services the interrupt request, the printer attachment presents:



Device Not Ready. The Output Print command will result in an interrupt presentation with bits 2 and 3 present in Status Word 1.

Not Attached/Off Line. When the printer attachment is under control of the CE panel or power to the 5024 is Off, Code 3 will be returned to the Initiate I/O command.

Operating and Programming Restrictions

If the printer is in the ready condition before a power interruption, it will be ready after the power disturbance. The belt image buffer will have to be reinitialized. The forms position will be lost if power is interrupted and will require operator intervention to realign the forms.

A reset device command will not alter the belt image buffer.

The completion status of the line being printed or spaced at the time of the system reset is unknown. Recovery of this status, if desired, will require operator intervention.

Switches and Indicators

Figure 17-10 illustrates the printer switches and indicators.

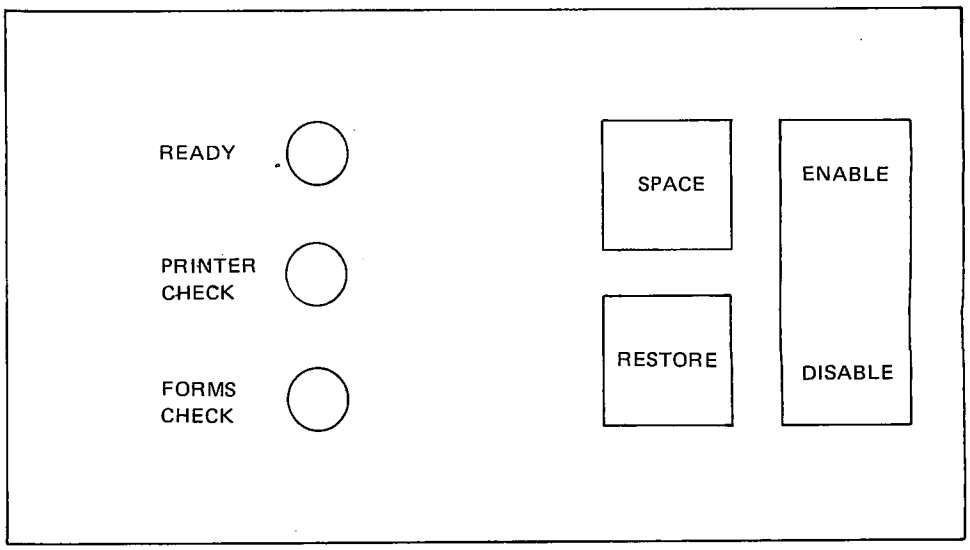


Figure 17-11. Printer switches and indicators

Power Switch

The power to the Printer Attachment is controlled by ON/OFF switch on the 5024 stand. In order for the printer to be ready, this switch must be ON.

PRINTER CHECK Light

Read forms parameter and examine bits 0–7 of the diagnostic status word to determine the error condition.

FORMS CHECK Indicators

Read forms parameter and examine bits 8–15 of the diagnostic status word to determine the error condition.

READY Indicator

This indicates that the printer is ready to execute system commands. The READY indicator can come on with no forms in the printer. It will go off after a print command is issued.

ENABLE/DISABLE Switch

This switch must be in the ENABLE position to enable system control of the printer. When the switch is in ENABLE and there are no check conditions, the ready indicator will be ON. If the switch is turned to DISABLE during a print operation, the printer will complete the current line and then stop. If the carriage is moving when the switch is also used to reset any error lights. Turn switch from ENABLE to DISABLE and back to ENABLE to reset errors.

SPACE Key

This key is active only when the printer is in the not ready state. Pressing the key causes the carriage to space the forms one line. This switch will also cause the electronic VFC to increment by one. To multiple space, hold this key down.

RESTORE Key

The carriage restore key causes the carriage to be positioned at line one when the printer is not ready. If the cover is open, only the electronic VFC will be set to line one. If the cover is closed, both the electronic VFC and the forms will move to line one. To multiple restore, hold this key down.

To align the forms, the operator manually adjusts the paper to the first line on the form then with the cover open, presses the carriage restore key. This adjusts the electronic VFC to line one. The cover is then closed and the printer made ready.

The instructions in this appendix are in alphabetical order by name. Use of the leading P in the instruction mnemonics is required in all System/7 programs to be assembled on an 1130 or 1800 system, or using the System/7 Stand-Alone Assembler. The P or an X is always required in the Execute I/O instruction (PIO or XIO). The P is optional (except for PIO) when assembling on a System/360, System/370, or the System/7 Macro Assembler.

Key to symbols for all tables in this appendix:

Symbol	Meaning
Acc	Accumulator
Addr	Contents of the address portion of a two-word instruction
Disp	Contents of the displacement portion of a one-word instruction
EA	Effective address
IAR	Instruction address register
XR...	Index register 1, 2, 3, 4, 5, 6, or 7

Add

Mnemonic	Execution time (ns)	Op code	R	Displacement	Hex equivalent	Description of instruction
[P] A	800	10000	000	XXXX XXXX	80XX	Add contents at EA (IAR+Disp) to Acc
[P] A	800	10000	001	XXXX XXXX	81XX	Add contents at EA (XR1+Disp) to Acc
[P] A	800	10000	010	XXXX XXXX	82XX	Add contents at EA (XR2+Disp) to Acc
[P] A	800	10000	011	XXXX XXXX	83XX	Add contents at EA (XR3+Disp) to Acc
[P] A	800	10000	100	XXXX XXXX	84XX	Add contents at EA (XR4+Disp) to Acc
[P] A	800	10000	101	XXXX XXXX	85XX	Add contents at EA (XR5+Disp) to Acc
[P] A	800	10000	110	XXXX XXXX	86XX	Add contents at EA (XR6+Disp) to Acc
[P] A	800	10000	111	XXXX XXXX	87XX	Add contents at EA (XR7+Disp) to Acc

Add immediate

Mnemonic	Execution time (ns)	Op code	R	Displacement	Hex equivalent	Description of instruction
[P] AI	400	01110	000	XXXX XXXX	70XX	Add expanded displacement to contents of Acc
[P] AI	400	01110	001	XXXX XXXX	71XX	Add expanded displacement to contents of XR1
[P] AI	400	01110	010	XXXX XXXX	72XX	Add expanded displacement to contents of XR2
[P] AI	400	01110	011	XXXX XXXX	73XX	Add expanded displacement to contents of XR3
[P] AI	400	01110	100	XXXX XXXX	74XX	Add expanded displacement to contents of XR4
[P] AI	400	01110	101	XXXX XXXX	75XX	Add expanded displacement to contents of XR5
[P] AI	400	01110	110	XXXX XXXX	76XX	Add expanded displacement to contents of XR6
[P] AI	400	01110	111	XXXX XXXX	77XX	Add expanded displacement to contents of XR7

Add register

Mnemonic	Execution time (ns)	Op code	R	Modifier	Hex equivalent	Description of instruction
[P] AR	400	11111	000	0000 0001	F801	Add contents of Acc to Acc
[P] AR	400	11111	001	0000 0001	F901	Add contents of XR1 to Acc
[P] AR	400	11111	010	0000 0001	FA01	Add contents of XR2 to Acc
[P] AR	400	11111	011	0000 0001	FB01	Add contents of XR3 to Acc
[P] AR	400	11111	100	0000 0001	FC01	Add contents of XR4 to Acc
[P] AR	400	11111	101	0000 0001	FD01	Add contents of XR5 to Acc
[P] AR	400	11111	110	0000 0001	FE01	Add contents of XR6 to Acc
[P] AR	400	11111	111	0000 0001	FF01	Add contents of XR7 to Acc

See key to symbols on page A-1

Add to storage and skip

Mnemonic	Execution time (ns)	Op code	R	Displacement	Hex equivalent	Description of instruction
[P] AS	1200	01111	000	XXXX XXXX	7 8 X X	Add 1 to contents at EA (IAR+Disp) and skip the next one word of instruction
[P] AS	1200	01111	001	XXXX XXXX	7 9 X X	Add 1 to contents at EA (XR1+Disp) and skip the next one word of instruction
[P] AS	1200	01111	010	XXXX XXXX	7 A X X	Add 1 to contents at EA (XR2+Disp) and skip the next one word of instruction
[P] AS	1200	01111	011	XXXX XXXX	7 B X X	Add 1 to contents at EA (XR3+Disp) and skip the next one word of instruction
[P] AS	1200	01111	100	XXXX XXXX	7 C X X	Add 1 to contents at EA (XR4+Disp) and skip the next one word of instruction
[P] AS	1200	01111	101	XXXX XXXX	7 D X X	Add 1 to contents at EA (XR5+Disp) and skip the next one word of instruction
[P] AS	1200	01111	110	XXXX XXXX	7 E X X	Add 1 to contents at EA (XR6+Disp) and skip the next one word of instruction
[P] AS	1200	01111	111	XXXX XXXX	7 F X X	Add 1 to contents at EA (XR7+Disp) and skip the next one word of instruction

Alter protect key (provided only with 5010 Processor Module Model E)

Mnemonic	Execution time (ns)	Op code	R	Fun	Modifier	Hex-equivalent	Description of Instruction
							Store bits 13 through 15 of the
APK	1200-1800*	00001	000	000	000000001	08001000†	ACC in the protect key for the current level
APK	1200-1800*	00001	001	000	000000001	09001000	XR1 in the protect key for the current level
APK	1200-1800*	00001	010	000	000000001	0A001000	XR2 in the protect key for the current level
APK	1200-1800*	00001	011	000	000000001	0B001000	XR3 in the protect key for the current level
APK	1200-1800*	00001	100	000	000000001	0C001000	XR4 in the protect key for the current level
APK	1200-1800*	00001	101	000	000000001	0D001000	XR5 in the protect key for the current level
APK	1200-1800*	00001	110	000	000000001	0E101000	XR6 in the protect key for the current level
APK	1200-1800*	00001	111	000	000000001	0F001000	XR7 in the protect key for the current level

*Interface delay times cause a range of execution times.
†The device and module addresses in this instruction are always zero.

See key to symbols on page A-1

Alter storage key (provided only with 5010 Processor Module Model E)

Mnemonic	Execution time (ns)	Op code	R	Fun	Modifier	Hex equivalent	Description of Instruction
ASK	1200-1800*	00001	000	000	000000000	08000000†	Store bits 13 through 15 of the Acc in the selected storage key register
ASK	1200-1800*	00001	001	000	000000000	09000000	XR1 in the selected storage key register
ASK	1200-1800*	00001	010	000	000000000	0A000000	XR2 in the selected storage key register
ASK	1200-1800*	00001	011	000	000000000	0B000000	XR3 in the selected storage key register
ASK	1200-1800*	00001	100	000	000000000	0C000000	XR4 in the selected storage key register
ASK	1200-1800*	00001	101	000	000000000	0D000000	XR5 in the selected storage key register
ASK	1200-1800*	00001	110	000	000000000	0E000000	XR6 in the selected storage key register
ASK	1200-1800*	00001	111	000	00000000	0F000000	XR7 in the selected storage key register

*Interface delay times cause a range of execution times.
†The device and module addresses in this instruction are always zero.

AND

Mnemonic	Execution time (ns)	Op code	R	Displacement	Hex equivalent	Description of instruction
[P] N	800	11100	000	XXXX XXXX	E 0 X X	AND contents at EA (IAR+Disp) with Acc
[P] N	800	11100	001	XXXX XXXX	E 1 X X	AND contents at EA (XR1+Disp) with Acc
[P] N	800	11100	010	XXXX XXXX	E 2 X X	AND contents at EA (XR2+Disp) with Acc
[P] N	800	11100	011	XXXX XXXX	E 3 X X	AND contents at EA (XR3+Disp) with Acc
[P] N	800	11100	100	XXXX XXXX	E 4 X X	AND contents at EA (XR4+Disp) with Acc
[P] N	800	11100	101	XXXX XXXX	E 5 X X	AND contents at EA (XR5+Disp) with Acc
[P] N	800	11100	110	XXXX XXXX	E 6 X X	AND contents at EA (XR6+Disp) with Acc
[P] N	800	11100	111	XXXX XXXX	E 7 X X	AND contents at EA (XR7+Disp) with Acc

See key to symbols on page A-1

AND register

Mnemonic	Execution time (ns)	Op code	R	Modifier	Hex equivalent	Description of instruction
[P] NR	400	11111	000	0000 0011	F 8 0 3	AND contents of Acc with Acc
[P] NR	400	11111	001	0000 0011	F 9 0 3	AND contents of XR1 with Acc
[P] NR	400	11111	010	0000 0011	F A 0 3	AND contents of XR2 with Acc
[P] NR	400	11111	011	0000 0011	F B 0 3	AND contents of XR3 with Acc
[P] NR	400	11111	100	0000 0011	F C 0 3	AND contents of XR4 with Acc
[P] NR	400	11111	101	0000 0011	F D 0 3	AND contents of XR5 with Acc
[P] NR	400	11111	110	0000 0011	F E 0 3	AND contents of XR6 with Acc
[P] NR	400	11111	111	0000 0011	F F 0 3	AND contents of XR7 with Acc

AND to mask

Mnemonic	Execution time (ns)	Op code	R	Modifier	Hex equivalent	Description of instruction
[P] NM	400	10110	000	0000 0000	B 0 0 0	AND contents of Acc with interruption mask register
[P] NM	400	10110	001	0000 0000	B 1 0 0	AND contents of XR1 with interruption mask register
[P] NM	400	10110	010	0000 0000	B 2 0 0	AND contents of XR2 with interruption mask register
[P] NM	400	10110	011	0000 0000	B 3 0 0	AND contents of XR3 with interruption mask register
[P] NM	400	10110	100	0000 0000	B 4 0 0	AND contents of XR4 with interruption mask register
[P] NM	400	10110	101	0000 0000	B 5 0 0	AND contents of XR5 with interruption mask register
[P] NM	400	10110	110	0000 0000	B 6 0 0	AND contents of XR6 with interruption mask register
[P] NM	400	10110	111	0000 0000	B 7 0 0	AND contents of XR7 with interruption mask register

See key to symbols on page A-1

Branch

Mnemonic	Execution time (ns)	Op code	R	Displacement	Hex equivalent	Description of instruction
[P] B	400	00111	000	XXXX XXXX	3 8 X X	Branch to location at EA (IAR+Disp)
[P] B	400	00111	001	XXXX XXXX	3 9 X X	Branch to location at EA (XR1+Disp)
[P] B	400	00111	010	XXXX XXXX	3 A X X	Branch to location at EA (XR2+Disp)
[P] B	400	00111	011	XXXX XXXX	3 B X X	Branch to location at EA (XR3+Disp)
[P] B	400	00111	100	XXXX XXXX	3 C X X	Branch to location at EA (XR4+Disp)
[P] B	400	00111	101	XXXX XXXX	3 D X X	Branch to location at EA (XR5+Disp)
[P] B	400	00111	110	XXXX XXXX	3 E X X	Branch to location at EA (XR6+Disp)
[P] B	400	00111	111	XXXX XXXX	3 F X X	Branch to location at EA (XR7+Disp)

Branch and link

Mnemonic	Execution time (ns)	Op code	R	Displacement	Hex equivalent	Description of instruction
[P] BAL	400	01011	000	XXXX XXXX	5 8 X X	Store contents of IAR in Acc and branch to location at EA (IAR+Disp)
[P] BAL	400	01011	001	XXXX XXXX	5 9 X X	Store contents of IAR in XR1 and branch to location at EA (IAR+Disp)
[P] BAL	400	01011	010	XXXX XXXX	5 A X X	Store contents of IAR in XR2 and branch to location at EA (IAR+Disp)
[P] BAL	400	01011	011	XXXX XXXX	5 B X X	Store contents of IAR in XR3 and branch to location at EA (IAR+Disp)
[P] BAL	400	01011	100	XXXX XXXX	5 C X X	Store contents of IAR in XR4 and branch to location at EA (IAR+Disp)
[P] BAL	400	01011	101	XXXX XXXX	5 D X X	Store contents of IAR in XR5 and branch to location at EA (IAR+Disp)
[P] BAL	400	01011	110	XXXX XXXX	5 E X X	Store contents of IAR in XR6 and branch to location at EA (IAR+Disp)
[P] BAL	400	01011	111	XXXX XXXX	5 F X X	Store contents of IAR in XR7 and branch to location at EA (IAR+Disp)

See key to symbols on page A-1

Branch and link long

Mnemonic	Execution time (ns)	Op code	R		Hex equivalent	Description of instruction
[P] BALL*	800	01010	000	0000 0000	5000	Store contents of IAR in Acc and branch to location at EA (Addr)
[P] BALL*	800	01010	001	0000 0000	5100	Store contents of IAR in XR1 and branch to location at EA (Addr)
[P] BALL*	800	01010	010	0000 0000	5200	Store contents of IAR in XR2 and branch to location at EA (Addr)
[P] BALL*	800	01010	011	0000 0000	5300	Store contents of IAR in XR3 and branch to location at EA (Addr)
[P] BALL*	800	01010	100	0000 0000	5400	Store contents of IAR in XR4 and branch to location at EA (Addr)
[P] BALL*	800	01010	101	0000 0000	5500	Store contents of IAR in XR5 and branch to location at EA (Addr)
[P] BALL*	800	01010	110	0000 0000	5600	Store contents of IAR in XR6 and branch to location at EA (Addr)
[P[BALL*	800	01010	111	0000 0000	5700	Store contents of IAR in XR7 and branch to location at EA (Addr)

*BALL is a two-word instruction; the second word contains the branch-to address.

Branch and unmask long (Provided only with 5010 Processor Module Model E)

Mnemonic	Execution time (ns)	Op code	R	Modifier	Hex equivalent	Description of instruction
BUL*	800	01010	000	00001000	5008	Reset summary mask, branch to location at EA (Addr)

*BUL is a two-word instruction; the second word contains the branch-to address.

See key to symbols on page A-1

Branch conditional

Mnemonic	Execution time (ns)	Op code	R	Condition bits	Hex equivalent	Description of instruction
[P] BC	400**	01000	000*	XXXX XXXX	40XX	Branch conditional to location at EA (Addr)
[P] BC	400**	01000	001	XXXX XXXX	41XX	Branch conditional to location at EA (XR1)
[P] BC	400**	01000	010	XXXX XXXX	42XX	Branch conditional to location at EA (XR2)
[P] BC	400**	01000	011	XXXX XXXX	43XX	Branch conditional to location at EA (XR3)
[P] BC	400**	01000	100	XXXX XXXX	44XX	Branch conditional to location at EA (XR4)
[P] BC	400**	01000	101	XXXX XXXX	45XX	Branch conditional to location at EX (XR5)
[P] BC	400**	01000	110	XXXX XXXX	46XX	Branch conditional to location at EA (XR6)
[P] BC	400**	01000	111	XXXX XXXX	47XX	Branch conditional to location at EA (XR7)

*This is a two-word instruction; the second word contains th branch-to address.

**If no branch is taken, BC execution time = 400 nanoseconds. It is 800 nanoseconds if the branch is taken.

Condition bits for BC instruction

Bit	Meaning or condition tested
8	Overflow save flag
9	Carry and overflow indicators both off (condition code 0)
10	Result-zero indicator on
11	Result-negative indicator on
12	Result-positive indicator on
13	Result-even indicator on
14	Carry indicator off
15	Overflow indicator off

See key to symbols on page A-1

Complement register

Mnemonic	Execution time (ns)	Op code	R	Modifier	Hex equivalent	Description of instruction
[P] CR	400	11111	000	0000 1000	F 8 0 8	Complement contents of Acc
[P] CR	400	11111	001	0000 1000	F 9 0 8	Complement contents of XR1
[P] CR	400	11111	010	0000 1000	F A 0 8	Complement contents of XR2
[P] CR	400	11111	011	0000 1000	F B 0 8	Complement contents of XR3
[P] CR	400	11111	100	0000 1000	F C 0 8	Complement contents of XR4
[P] CR	400	11111	101	0000 1000	F D 0 8	Complement contents of XR5
[P] CR	400	11111	110	0000 1000	F E 0 8	Complement contents of XR6
[P] CR	400	11111	111	0000 1000	F F 0 8	Complement contents of XR7

Exclusive OR

Mnemonic	Execution time (ns)	Op code	R	Displacement	Hex equivalent	Description of instruction
[P] X	800	11110	000	XXXX XXXX	F 0 X X	Exclusive OR contents at EA (IAR+Disp) with Acc
[P] X	800	11110	001	XXXX XXXX	F 1 X X	Exclusive OR contents at EA (XR1+Disp) with Acc
[P] X	800	11110	010	XXXX XXXX	F 2 X X	Exclusive OR contents at EA (XR2+Disp) with Acc
[P] X	800	11110	011	XXXX XXXX	F 3 X X	Exclusive OR contents at EA (XR3+Disp) with Acc
[P] X	800	11110	100	XXXX XXXX	F 4 X X	Exclusive OR contents at EA (XR4+Disp) with Acc
[P] X	800	11110	101	XXXX XXXX	F 5 X X	Exclusive OR contents at EA (XR5+Disp) with Acc
[P] X	800	11110	110	XXXX XXXX	F 6 X X	Exclusive OR contents at EA (XR6+Disp) with Acc
[P] X	800	11110	111	XXXX XXXX	F 7 X X	Exclusive OR contents at EA (XR7+Disp) with Acc

See key to symbols on page A-1

Exclusive OR register

Mnemonic	Execution time (ns)	Op code	R	Modifier	Hex equivalent	Description of instruction
[P] XR	400	11111	000	0000 0101	F 8 0 5	Exclusive OR contents of Acc with Acc
[P] XR	400	11111	001	0000 0101	F 9 0 5	Exclusive OR contents of XR1 with Acc
[P] XR	400	11111	010	0000 0101	F A 0 5	Exclusive OR contents of XR2 with Acc
[P] XR	400	11111	011	0000 0101	F B 0 5	Exclusive OR contents of XR3 with Acc
[P] XR	400	11111	100	0000 0101	F C 0 5	Exclusive OR contents of XR4 with Acc
[P] XR	400	11111	101	0000 0101	F D 0 5	Exclusive OR contents of XR5 with Acc
[P] XR	400	11111	110	0000 0101	F E 0 5	Exclusive OR contents of XR6 with Acc
[P] XR	400	11111	111	0000 0101	F F 0 5	Exclusive OR contents of XR7 with Acc

Execute I/O

Mnemonic	Execution time (ns)	Op code	R	Fun		Hex equivalent	Description of instruction
PIO*	1800**	00001	000	XXX	0 0000	0 8 X 0	Execute I/O operation using accumulator contents for data
PIO*	1800**	00001	001	XXX	0 0000	0 9 X 0	Execute I/O operation using contents of XR1 for data

*PIO is a two-word instruction. See instruction description for more detail. An X may be substituted for the P.

**Execution time for an immediate operation = (1800+D) ns, where D = delay. This delay depends on the distance between the I/O module and the processor module and will range from 100 to 2100 nanoseconds. For system configurations that do not include a 5026 Enclosure Model D3 or D6, the delay will range from 100 to 800 nanoseconds.

See key to symbols on page A-1

Inspect IAR backup

Mnemonic	Execution time (ns)	Op code	R		Level	Modifier	Hex equivalent	Description of instruction
[P]IIB	800	11111	000	00	XX	1100	F 8 X C	Load contents of backup IAR into Acc and reset the selected level
[P]IIB	800	11111	001	00	XX	1100	F 9 X C	Load contents of backup IAR into XR1 and reset the selected level
[P]IIB	800	11111	010	00	XX	1100	F A X C	Load contents of backup IAR into XR2 and reset the selected level
[P]IIB	800	11111	011	00	XX	1100	F B X C	Load contents of backup IAR into XR3 and reset the selected level
[P]IIB	800	11111	100	00	XX	1100	F C X C	Load contents of backup IAR into XR4 and reset the selected level
[P]IIB	800	11111	101	00	XX	1100	F D X C	Load contents of backup IAR into XR5 and reset the selected level
[P]IIB	800	11111	110	00	XX	1100	F E X C	Load contents of backup IAR into XR6 and reset the selected level
[P]IIB	800	11111	111	00	XX	1100	F F X C	Load contents of backup IAR into XR7 and reset the selected level

Interchange register

Mnemonic	Execution time (ns)	Op code	R	Modifier	Hex equivalent	Description of instruction
[P]IR	400	11111	000	0000 1010	F 8 0 A	Exchange contents of Acc with contents of Acc
[P]IR	400	11111	001	0000 1010	F 9 0 A	Exchange contents of Acc with contents of XR1
[P]IR	400	11111	010	0000 1010	F A 0 A	Exchange contents of Acc with contents of XR2
[P]IR	400	11111	011	0000 1010	F B 0 A	Exchange contents of Acc with contents of XR3
[P]IR	400	11111	100	0000 1010	F C 0 A	Exchange contents of Acc with contents of XR4
[P]IR	400	11111	101	0000 1010	F D 0 A	Exchange contents of Acc with contents of XR5
[P]IR	400	11111	110	0000 1010	F E 0 A	Exchange contents of Acc with contents of XR6
[P]IR	400	11111	111	0000 1010	F F 0 A	Exchange contents of Acc with contents of XR7

See key to symbols on page A-1

Level exit

Mnemonic	Execution time (ns)	Op code	R		Hex equivalent	Description of instruction
[P] LEX	400	00110	000	0000 0000	3 0 0 0	Exit from current priority level

Load accumulator

Mnemonic	Execution time (ns)	Op code	R	Displacement	Hex equivalent	Description of instruction
[P] L	800	11000	000	XXXX XXXX	C 0 X X	Contents of storage at EA (IAR+Disp) are loaded into Acc
[P] L	800	11000	001	XXXX XXXX	C 1 X X	Contents of storage at EA (XR1+Disp) are loaded into Acc
[P] L	800	11000	010	XXXX XXXX	C 2 X X	Contents of storage at EA (XR2+Disp) are loaded into Acc
[P] L	800	11000	011	XXXX XXXX	C 3 X X	Contents of storage at EA (XR3+Disp) are loaded into Acc
[P] L	800	11000	100	XXXX XXXX	C 4 X X	Contents of storage at EA (XR4+Disp) are loaded into Acc
[P] L	800	11000	101	XXXX XXXX	C 5 X X	Contents of storage at EA (XR5+Disp) are loaded into Acc
[P] L	800	11000	110	XXXX XXXX	C 6 X X	Contents of storage at EA (XR6+Disp) are loaded into Acc
[P] L	800	11000	111	XXXX XXXX	C 7 X X	Contents of storage at EA (XR7+Disp) are loaded into Acc

See key to symbols on page A-1

Load and zero

Mnemonic	Execution time (ns)	Op code	R	Displacement	Hex equivalent	Description of instruction
[P] LZ	1200	11001	000	XXXX XXXX	C 8 X X	Contents of storage at EA (IAR+Disp) loaded into Acc and contents at EA set to 0
[P] LZ	1200	11001	001	XXXX XXXX	C 9 X X	Contents of storage at EA (XR1+Disp) loaded into Acc and contents of EA set to 0
[P] LZ	1200	11001	010	XXXX XXXX	C A X X	Contents of storage at EA (XR2+Disp) loaded into Acc and contents at EA set to 0
[P] LZ	1200	11001	011	XXXX XXXX	C B X X	Contents of storage at EA (XR3+Disp) loaded into Acc and contents at EA set to 0
[P] LZ	1200	11001	100	XXXX XXXX	C C X X	Contents of storage at EA (XR4+Disp) loaded into Acc and contents at EA set to 0
[P] LZ	1200	11001	101	XXXX XXXX	C D X X	Contents of storage at EA (XR5+Disp) loaded into Acc and contents at EA set to 0
[P] LZ	1200	11001	110	XXXX XXXX	C E X X	Contents of storage at EA (XR6+Disp) loaded into Acc and contents at EA set to 0
[P] LZ	1200	11001	111	XXXX XXXX	C F X X	Contents of storage at EA (XR7+Disp) loaded into Acc and contents at EA set to 0

Load from register

Mnemonic	Execution time (ns)	Op code	R	Modifier	Hex equivalent	Description of instruction
[P] LR	400	11111	000	0000 0111	F 8 0 7	Load contents of IAR into Acc
[P] LR	400	11111	001	0000 0111	F 9 0 7	Load contents of XR1 into Acc
[P] LR	400	11111	010	0000 0111	F A 0 7	Load contents of XR2 into Acc
[P] LR	400	11111	011	0000 0111	F B 0 7	Load contents of XR3 into Acc
[P] LR	400	11111	100	0000 0111	F C 0 7	Load contents of XR4 into Acc
[P] LR	400	11111	101	0000 0111	F D 0 7	Load contents of XR5 into Acc
[P] LR	400	11111	110	0000 0111	F E 0 7	Load contents of XR6 into Acc
[P] LR	400	11111	111	0000 0111	F F 0 7	Load contents of XR7 into Acc

See key to symbols on page A-1

Load immediate

Mnemonic	Execution time (ns)	Op code	R	Displacement	Hex equivalent	Description of instruction
[P] LI	400	01100	000	XXXX XXXX	6 0 X X	Load expanded displacement into Acc
[P] LI	400	01100	001	XXXX XXXX	6 1 X X	Load expanded displacement into XR1
[P] LI	400	01100	010	XXXX XXXX	6 2 X X	Load expanded displacement into XR2
[P] LI	400	01100	011	XXXX XXXX	6 3 X X	Load expanded displacement into XR3
[P] LI	400	01100	100	XXXX XXXX	6 4 X X	Load expanded displacement into XR4
[P] LI	400	01100	101	XXXX XXXX	6 5 X X	Load expanded displacement into XR5
[P] LI	400	01100	110	XXXX XXXX	6 6 X X	Load expanded displacement into XR6
[P] LI	400	01100	111	XXXX XXXX	6 7 X X	Load expanded displacement into XR7

Load index long

Mnemonic	Execution time (ns)	Op code	R1	R2		Hex equivalent	Description of instruction
[P] LXL*	1200	10001	000	000	0 0000	8 8 0 0	Contents of storage at EA (Addr) loaded into Acc
[P] LXL*	1200	10001	001	000	0 0000	8 9 0 0	Contents of storage at EA (Addr) loaded into XR1
[P] LXL*	1200	10001	010	001	0 0000	8 A 2 0	Contents of storage at EA (Addr+XR1) loaded into XR2
[P] LXL*	1200	10001	011	010	0 0000	8 B 4 0	Contents of storage at EA (Addr+XR2) loaded into XR3
[P] LXL*	1200	10001	101	111	0 0000	8 D E 0	Contents of storage at EA (Addr+XR7) loaded into XR5

*LXL is a two-word instruction. Any combination of valid values for R1 and R2 can be used.

See key to symbols on page A-1

Load processor status

Mnemonic	Execution time (ns)	Op code	R	Modifier	Hex equivalent	Description of instruction
[P] LPS	400	11111	000	0000 1011	F 8 0 B	Load processor status into Acc
[P] LPS	400	11111	001	0000 1011	F 9 0 B	Load processor status into XR1
[P] LPS	400	11111	010	0000 1011	F A 0 B	Load processor status into XR2
[P] LPS	400	11111	011	0000 1011	F B 0 B	Load processor status into XR3
[P] LPS	400	11111	100	0000 1011	F C 0 B	Load processor status into XR4
[P] LPS	400	11111	101	0000 1011	F D 0 B	Load processor status into XR5
[P] LPS	400	11111	110	0000 1011	F E 0 B	Load processor status into XR6
[P] LPS	400	11111	111	0000 1011	F F 0 B	Load processor status into XR7

OR

Mnemonic	Execution time (ns)	Op code	R	Displacement	Hex equivalent	Description of instruction
[P] O	800	11101	000	XXXX XXXX	E 8 X X	OR contents at EA (IAR+Disp) with Acc
[P] O	800	11101	001	XXXX XXXX	E 9 X X	OR contents at EA (XR1+Disp) with Acc
[P] O	800	11101	010	XXXX XXXX	E A X X	OR contents at EA (XR2+Disp) with Acc
[P] O	800	11101	011	XXXX XXXX	E B X X	OR contents at EA (XR3+Disp) with Acc
[P] O	800	11101	100	XXXX XXXX	E C X X	OR contents at EA (XR4+Disp) with Acc
[P] O	800	11101	101	XXXX XXXX	E D X X	OR contents at EA (XR5+Disp) with Acc
[P] O	800	11101	110	XXXX XXXX	E E X X	OR contents at EA (XR6+Disp) with Acc
[P] O	800	11101	111	XXXX XXXX	E F X X	OR contents at EA (XR7+Disp) with Acc

See key to symbols on page A-1

OR register

Mnemonic	Execution time (ns)	Op code	R	Modifier	Hex equivalent	Description of instruction
[P]OR	400	11111	000	0000 0100	F 8 0 4	OR contents of Acc with Acc
[P]OR	400	11111	001	0000 0100	F 9 0 4	OR contents of XR1 with Acc
[P]OR	400	11111	010	0000 0100	F A 0 4	OR contents of XR2 with Acc
[P]OR	400	11111	011	0000 0100	F B 0 4	OR contents of XR3 with Acc
[P]OR	400	11111	100	0000 0100	F C 0 4	OR contents of XR4 with Acc
[P]OR	400	11111	101	0000 0100	F D 0 4	OR contents of XR5 with Acc
[P]OR	400	11111	110	0000 0100	F E 0 4	OR contents of XR6 with Acc
[P]OR	400	11111	111	0000 0100	F F 0 4	OR contents of XR7 with Acc

OR to mask

Mnemonic	Execution time (ns)	Op code	R	Modifier	Hex equivalent	Description of instruction
[P]OM	400	10110	000	0000 0001	B 0 0 1	OR contents of Acc with interruption mask register
[P]OM	400	10110	001	0000 0001	B 1 0 1	OR contents of XR1 with interruption mask register
[P]OM	400	10110	010	0000 0001	B 2 0 1	OR contents of XR2 with interruption mask register
[P]OM	400	10110	011	0000 0001	B 3 0 1	OR contents of XR3 with interruption mask register
[P]OM	400	10110	100	0000 0001	B 4 0 1	OR contents of XR4 with interruption mask register
[P]OM	400	10110	101	0000 0001	B 5 0 1	OR contents of XR5 with interruption mask register
[P]OM	400	10110	110	0000 0001	B 6 0 1	OR contents of XR6 with interruption mask register
[P]OM	400	10110	111	0000 0001	B 7 0 1	OR contents of XR7 with interruption mask register

See key to symbols on page A-1

Read IAR backup (Provided only with 5010 Processor Module Model E)

Mnemonic	Execution time (ns)	Op code	R		Level	Mod-ifier	Hex equi-valent	Description of instruction
RIB	800	11111	000	00	XX	1101	F 8 X D	Load contents of backup IAR into Acc but do not reset selected level
RIB	800	11111	001	00	XX	1101	F 9 X D	Load contents of backup IAR into XR1 but do not reset selected level
RIB	800	11111	010	00	XX	1101	F A X D	Load contents of backup IAR into XR2 but do not reset selected level
RIB	800	11111	011	00	XX	1101	F B X D	Load contents of backup IAR into XR3 but do not reset selected level
RIB	800	11111	100	00	XX	1101	F C X D	Load contents of backup IAR into XR4 but do not reset selected level
RIB	800	11111	101	00	XX	1101	F D X D	Load contents of backup IAR into XR5 but do not reset selected level
RIB	800	11111	110	00	XX	1101	F E X D	Load contents of backup IAR into XR6 but do not reset selected level
RIB	800	11111	111	00	XX	1101	F F X D	Load contents of backup IAR into XR7 but do not reset selected level

See key to symbols on page A-1

Read protect key (provided only with 5010 Processor Module Model E)

Mnemonic	Execution time (ns)	Op code	R	Fun	Modifier	Hex equivalent	Description of Instruction
							Replace bits 13 through 15 of:
RPK	1200-1800*	00001	000	000	000000010	08002000†	Acc with contents of the protect key
RPK	1200-1800*	00001	001	000	000000010	09002000†	XR1 with contents of the protect key
RPK	1200-1800*	00001	010	000	000000010	0A002000†	XR2 with contents of the protect key
RPK	1200-1800*	00001	011	000	000000010	0B002000†	XR3 with contents of the protect key
RPK	1200-1800*	00001	100	000	000000010	0C002000†	XR4 with contents of the protect key
RPK	1200-1800*	00001	101	000	000000010	0D002000†	XR5 with contents of the protect key
RPK	1200-1800*	00001	110	000	000000010	0E002000†	XR6 with contents of the protect key
RPK	1200-1800*	00001	111	000	000000010	0F002000†	XR7 with contents of the protect key

*Interface delay times cause a range of execution times.
†The device and module addresses in this instruction are always zero.

See key to symbols on page A-1

Sense level and mask

Mnemonic	Execution time (ns)	Op code	R	Modifier	Hex equivalent	Description of instruction
[P]SLM	400	11111	000	000X* 1001	F 8 X 9	Load the active level number and contents of mask register into Acc
[P]SLM	400	11111	001	000X 1001	F 9 X 9	Load the active level number and contents of mask register into XR1
[P]SLM	400	11111	010	000X 1001	F A X 9	Load the active level number and contents of mask register into XR2
[P]SLM	400	11111	011	000X 1001	F B X 9	Load the active level number and contents of mask register into XR3
[P]SLM	400	11111	100	000X 1001	F C X 9	Load the active level number and contents of mask register into XR4
[P]SLM	400	11111	101	000X 1001	F D X 9	Load the active level number and contents of mask register into XR5
[P]SLM	400	11111	110	000X 1001	F E X 9	Load the active level number and contents of mask register into XR6
[P]SLM	400	11111	111	000X 1001	F F X 9	Load the active level number and contents of mask register into XR7

*See description of SLM instruction for use of bit 11 as a temporary mask.

Shift left circular

Mnemonic	Execution time (ns)	Op code	R	Mod-ifier	Count	Hex equi-valent	Description of instruction
[P]SLC	400*	00010	000	000	X XXXX	1 0 X X	Contents of Acc rotate left number of counts in Disp
[P]SLC	400*	00010	001	000	X XXXX	1 1 X X	Contents of XR1 rotate left number of counts in Disp
[P]SLC	400*	00010	010	000	X XXXX	1 2 X X	Contents of XR2 rotate left number of counts in Disp
[P]SLC	400*	00010	011	000	X XXXX	1 3 X X	Contents of XR3 rotate left number of counts in Disp
[P]SLC	400*	00010	100	000	X XXXX	1 4 X X	Contents of XR4 rotate left number of counts in Disp
[P]SLC	400*	00010	101	000	X XXXX	1 5 X X	Contents of XR5 rotate left number of counts in Disp
[P]SLC	400*	00010	110	000	X XXXX	1 6 X X	Contents of XR6 rotate left number of counts in Disp
[P]SLC	400*	00010	111	000	X XXXX	1 7 X X	Contents of XR7 rotate left number of counts in Disp

*Add to execution time: 50N + [50], where N = number of bits shifted, and [50] = added time for odd number of shift counts.

See key to symbols on page A-1

Shift left logical

Mnemonic	Execution time (ns)	Op code	R	Mod-ifier	Count	Hex equi-valent	Description of instruction
[P]SLL	400*	00010	000	001	X XXXX	1 0 X X	Contents of Acc shift left number of counts in Disp
[P]SLL	400*	00010	001	001	X XXXX	1 1 X X	Contents of XR1 shift left number of counts in Disp
[P]SLL	400*	00010	010	001	X XXXX	1 2 X X	Contents of XR2 shift left number of counts in Disp
[P]SLL	400*	00010	011	001	X XXXX	1 3 X X	Contents of XR3 shift left number of counts in Disp
[P]SLL	400*	00010	100	001	X XXXX	1 4 X X	Contents of XR4 shift left number of counts in Disp
[P]SLL	400*	00010	101	001	X XXXX	1 5 X X	Contents of XR5 shift left number of counts in Disp
[P[SLL	400*	00010	110	001	X XXXX	1 6 X X	Contents of XR6 shift left number of counts in Disp
[P]SLL	400*	00010	111	001	X XXXX	1 7 X X	Contents of XR7 shift left number of counts in Disp

*Add to execution time: 50N + [50], where N = number of bits shifted, and [50] = added time for odd number of shift counts.

Shift right arithmetic

Mnemonic	Execution time (ns)	Op code	R	Mod-ifier	Count	Hex equi-valent	Description of instruction
[P]SRA	400*	00010	000	011	X XXXX	1 0 X X	Contents of Acc shift right number of counts in Disp and sign bit value shifted in
[P]SRA	400*	00010	001	011	X XXXX	1 1 X X	Contents of XR1 shift right number of counts in Disp and sign bit value shifted in
[P]SRA	400*	00010	010	011	X XXXX	1 2 X X	Contents of XR2 shift right number of counts in Disp and sign bit value shifted in
[P]SRA	400*	00010	011	011	X XXXX	1 3 X X	Contents of XR3 shift right number of counts in Disp and sign bit value shifted in
[P]SRA	400*	00010	100	011	X XXXX	1 4 X X	Contents of XR4 shift right number of counts in Disp and sign bit value shifted in
[P]SRA	400*	00010	101	011	X XXXX	1 5 X X	Contents of XR5 shift right number of counts in Disp and sign bit value shifted in
[P]SRA	400*	00010	110	011	X XXXX	1 6 X X	Contents of XR6 shift right number of counts in Disp and sign bit value shifted in
[P]SRA	400*	00010	111	011	X XXXX	1 7 X X	Contents of XR7 shift right number of counts in Disp and sign bit value shifted in

*Add to execution time: 50N + [50], where N = number of bits shifted, and [50] = added time for odd number of shift counts.

See key to symbols on page A-1

Shift right logical

Mnemonic	Execution time (ns)	Op code	R	Mod-ifier	Count	Hex equi-valent	Description of instruction
[P]SRL	400*	00010	000	010	X XXXX	1 0 X X	Contents of Acc shift right number of counts in Disp
[P]SRL	400*	00010	001	010	X XXXX	1 1 X X	Contents of XR1 shift right number of counts in Disp
[P]SRL	400*	00010	010	010	X XXXX	1 2 X X	Contents of XR2 shift right number of counts in Disp
[P]SRL	400*	00010	011	010	X XXXX	1 3 X X	Contents of XR3 shift right number of counts in Disp
[P]SRL	400*	00010	100	010	X XXXX	1 4 X X	Contents of XR4 shift right number of counts in Disp
[P]SRL	400*	00010	101	010	X XXXX	1 5 X X	Contents of XR5 shift right number of counts in Disp
[P]SRL	400*	00010	110	010	X XXXX	1 6 X X	Contents of XR6 shift right number of counts in Disp
[P]SRL	400*	00010	111	010	X XXXX	1 7 X X	Contents of XR7 shift right number of counts in Disp

*Add to execution time: 50N + [50], where N = number of bits shifted, and [50] = added time for odd number of shift counts.

Skip conditional

Mnemonic	Execution time (ns)	Op code	R	Condition bits	Hex equivalent	Description of instruction
[P]SKC	400	01001	000	XXXX XXXX	4 8 X X	Skip conditional the next one word of instruction

Condition bits for SKC instruction

Bit	Meaning or condition tested
8	Overflow save flag
9	Carry and overflow indicators both off (condition code 0)
10	Result-zero indicator on
11	Result-negative indicator on
12	Result-positive indicator on
13	Result-even indicator on
14	Carry indicator off
15	Overflow indicator off

See key to symbols on page A-1

Stop

Mnemonic	Execution time (ns)	Op code	R		Hex equivalent	Description of instruction
[P]STP	400	00100	000	0000 0000	2000	Enter stop state

Store accumulator

Mnemonic	Execution time (ns)	Op code	R	Displacement	Hex equivalent	Description of instruction
[P]ST	800	11010	000	XXXX XXXX	D0XX	Store contents of Acc in location at EA (IAR+Disp)
[P]ST	800	11010	001	XXXX XXXX	D1XX	Store contents of Acc in location at EA (XR1+Disp)
[P]ST	800	11010	010	XXXX XXXX	D2XX	Store contents of Acc in location at EA (XR2+Disp)
[P]ST	800	11010	011	XXXX XXXX	D3XX	Store contents of Acc in location at EA (XR3+Disp)
[P]ST	800	11010	100	XXXX XXXX	D4XX	Store contents of Acc in location at EA (XR4+Disp)
[P]ST	800	11010	101	XXXX XXXX	D5XX	Store contents of Acc in location at EA (XR5+Disp)
[P]ST	800	11010	110	XXXX XXXX	D6XX	Store contents of Acc in location at EA (XR6+Disp)
[P]ST	800	11010	111	XXXX XXXX	D7XX	Store contents of Acc in location at EA (XR7+Disp)

Store index

Mnemonic	Execution time (ns)	Op code	R	Displacement	Hex equivalent	Description of instruction
[P]STX	800	01101	000	XXXX XXXX	68XX	Store 0's in location at EA (IAR+Disp)
[P]STX	800	01101	001	XXXX XXXX	69XX	Store contents of XR1 in location at EA(IAR+Disp)
[P]STX	800	01101	010	XXXX XXXX	6AXX	Store contents of XR2 in location at EA (IAR+Disp)
[P]STX	800	01101	011	XXXX XXXX	6BXX	Store contents of XR3 in location at EA (IAR+Disp)
[P]STX	800	01101	100	XXXX XXXX	6CXX	Store contents of XR4 in location at EA (IAR+Disp)
[P]STX	800	01101	101	XXXX XXXX	6DXX	Store contents of XR5 in location at EA (IAR+Disp)
[P]STX	800	01101	110	XXXX XXXX	6EXX	Store contents of XR6 in location at EA (IAR+Disp)
[P]STX	800	01101	111	XXXX XXXX	6FXX	Store contents of XR7 in location at EA (IAR+Disp)

See key to symbols on page A-1

Store indicators (Provided only with 5010 Processor Module Model E)

Mnemonic	Execution time (ns)	Op code	R		Level	Modifier	Hex equivalent	Description of instruction
STI	400	11111	000	00	XX	0000	F 8 X 0	Load contents of result, carry, and overflow indicators on selected level into Acc
STI	400	11111	001	00	XX	0000	F 9 X 0	Load contents of result, carry, and overflow indicators on selected level into XR1
STI	400	11111	010	00	XX	0000	F A X 0	Load contents of result, carry, and overflow indicators on selected level into XR2
STI	400	11111	011	00	XX	0000	F B X 0	Load contents of result, carry, and overflow indicators on selected level into XR3
STI	400	11111	100	00	XX	0000	F C X 0	Load contents of result, carry, and overflow indicators on selected level into XR4
STI	400	11111	101	00	XX	0000	F D X 0	Load contents of result, carry, and overflow indicators on selected level into XR5
STI	400	11111	110	00	XX	0000	F E X 0	Load contents of result, carry, and overflow indicators on selected level into XR6
STI	400	11111	111	00	XX	0000	F F X 0	Load contents of result, carry, and overflow indicators on selected level into XR7

Store to register

Mnemonic	Execution time (ns)	Op code	R	Modifier	Hex equivalent	Description of instruction
[P]STR	400	11111	000	0000 0110	F 8 0 6	Store contents of Acc into IAR
[P]STR	400	11111	001	0000 0110	F 9 0 6	Store contents of Acc into XR1
[P]STR	400	11111	010	0000 0110	F A 0 6	Store contents of Acc into XR2
[P]STR	400	11111	011	0000 0110	F B 0 6	Store contents of Acc into XR3
[P]STR	400	11111	100	0000 0110	F C 0 6	Store contents of Acc into XR4
[P]STR	400	11111	101	0000 0110	F D 0 6	Store contents of Acc into XR5
[P]STR	400	11111	110	0000 0110	F E 0 6	Store contents of Acc into XR6
[P]STR	400	11111	111	0000 0110	F F 0 6	Store contents of Acc into XR7

See key to symbols on page A-1

Subtract

Mnemonic	Execution time (ns)	Op code	R	Displacement	Hex equivalent	Description of instruction
[P]S	800	10010	000	XXXX XXXX	90XX	Subtract contents at EA (IAR+Disp) from Acc
[P]S	800	10010	001	XXXX XXXX	91XX	Subtract contents at EA (XR1+Disp) from Acc
[P]S	800	10010	010	XXXX XXXX	92XX	Subtract contents at EA (XR2+Disp) from Acc
[P]S	800	10010	011	XXXX XXXX	93XX	Subtract contents at EA (XR3+Disp) from Acc
[P]S	800	10010	100	XXXX XXXX	94XX	Subtract contents at EA (XR4+Disp) from Acc
[P]S	800	10010	101	XXXX XXXX	95XX	Subtract contents at EA (XR5+Disp) from Acc
[P]S	800	10010	110	XXXX XXXX	96XX	Subtract contents at EA (XR6+Disp) from Acc
[P]S	800	10010	111	XXXX XXXX	97XX	Subtract contents at EA (XR7+Disp) from Acc

Subtract register

Mnemonic	Execution time (ns)	Op code	R	Modifier	Hex equivalent	Description of instruction
[P]SR	400	11111	000	0000 0010	F802	Subtract contents of Acc from Acc
[P]SR	400	11111	001	0000 0010	F902	Subtract contents of XR1 from Acc
[P]SR	400	11111	010	0000 0010	FA02	Subtract contents of XR2 from Acc
[P]SR	400	11111	011	0000 0010	FB02	Subtract contents of XR3 from Acc
[P]SR	400	11111	100	0000 0010	FC02	Subtract contents of XR4 from Acc
[P]SR	400	11111	101	0000 0010	FD02	Subtract contents of XR5 from Acc
[P]SR	400	11111	110	0000 0010	FE02	Subtract contents of XR6 from Acc
[P]SR	400	11111	111	0000 0010	FF02	Subtract contents of XR7 from Acc

Supervisor call

Mnemonic	Execution time (ns)	Op code	Hex equivalent	Description of instruction
SVC	2000	10011	98XX *	Turn on summary mask and branch via contents of main storage location hex 0009

* Reserved

See key to symbols on page A-1

Write IAR backup (Provided only with 5010 Processor Module Model E)

Mnemonic	Execution time (ns)	Op code	R		Level	Mod-ifier	Hex equi-valent	Description of instruction
WIB	800	11111	000	00	XX	1110	F 8 X E	Load contents of Acc into backup IAR
WIB	800	11111	001	00	XX	1110	F 9 X E	Load contents of XR1 into backup IAR
WIB	800	11111	010	00	XX	1110	F A X E	Load contents of XR2 into backup IAR
WIB	800	11111	011	00	XX	1110	F B X E	Load contents of XR3 into backup IAR
WIB	800	11111	100	00	XX	1110	F C X E	Load contents of XR4 into backup IAR
WIB	800	11111	101	00	XX	1110	F D X E	Load contents of XR5 into backup IAR
WIB	800	11111	110	00	XX	1110	F E X E	Load contents of XR6 into backup IAR
WIB	800	11111	111	00	XX	1110	F F X E	Load contents of XR7 into backup IAR

See key to symbols on page A-1

System/7 I/O commands

PIO instruction	Function	Modifier	Device address	
			Subaddress	Zeros
Interval timers (X = timer)				
Immediate write	0 0 1			
Set timer 0 or 1		0 0 0 0	0 0 0 X	0 0
Stop timer 0 or 1		1 0 0 0		
Start timer 0 or 1		1 0 0 1		
Immediate read	0 1 0			
Read timer 0 or 1		1 0 0 0	0 0 0 X	0 0
ACCA				
Immediate write	0 0 1			
Diagnostic write		0 0 0 0	0 0 1 1	0 0
Transmit character		0 0 0 1		
Reset ACCA		1 0 0 0		
Transmit control		1 0 0 1		
Immediate read	0 1 0			
Read ISW		0 0 1 1	0 0 1 1	0 0
Read character		1 0 0 0		
BSCA				
Immediate write	0 0 1			
Write address		0 0 1 1	0 0 1 1	0 0
Initiate I/O		0 1 1 1		
Write diagnostic control		1 0 1 1		
Reset BSCA		1 1 1 1		
Immediate read	0 1 0			
Read ISW		0 0 1 1	0 0 1 1	0 0
Read ISW extension		1 0 1 1		
Read residual address		1 1 1 1		
5028 Operator Station				
Immediate write	0 0 1			
Turn-off motor and lock keyboard		0 0 0 0	0 0 1 0	0 0
Print only		0 0 1 0		
Punch only		0 1 0 0		
Print and punch		0 1 1 0		
Turn-on motor		1 0 0 0		
Keyboard entry and no print		1 0 0 1		
Keyboard entry and print		1 0 1 1		
Feed tape and no print		1 1 0 0		
Feed tape and print		1 1 1 0		
Immediate read	0 1 0			
Read ISW		0 0 1 1	0 0 1 0	0 0
Read character with tape feed		1 0 0 0		
Read character without tape feed		1 1 0 0		

PIO instruction	Function	Modifier	Device address	
			Subaddress	Zeros
Analog input model B1				
Immediate write	0 0 1			0 0
Convert analog input		0 0 0 0		
Convert analog input with external synchronization		0 0 0 1		
Immediate read	0 1 0			0 0
Read ADC		0 0 0 0		
Read ADC extended precision		0 0 0 1		
Read ISW		0 0 1 1		
Analog input model C1				
Immediate write	0 0 1			
Convert analog input		0 0 0 0	1 0 0 1	0 0
Convert analog input with external synchronization		0 0 0 1		
Immediate read	0 1 0			
Read ADC		0 0 0 0	1 0 0 1	0 0
Read ADC extended precision		0 0 0 1		
Read and convert ADC		0 0 1 0		
Read ISW		0 0 1 1		
Analog input model D1				
Immediate write	0 0 1			
Convert analog input		0 0 0 0	1 0 0 1	0 0
Convert analog input with external synchronization		0 0 0 1		
Immediate read	0 1 0			
Read ADC		0 0 0 0	1 0 0 1	0 0
Read ISW		0 0 1 1		
Digital input control (X = group)				
Immediate write	0 0 1			
Set digital input latch control (X=group)		0 0 0 0	0 X X X	0 0
Set test signal (X=group)		0 0 0 1	0 X X X	0 0
Set digital input reference reg (X=group)		0 0 1 0	0 0 0 X	0 0
Set digital input interrupt control		0 0 1 1	0 0 0 X	0 0
Immediate read	0 1 0			
Read digital input register		0 0 0 0	0 X X X	0 0
Read/reset digital input register		0 0 0 1	0 X X X	0 0
Read digital input reference register		0 0 1 0	0 0 0 X	0 0
Read ISW		0 0 1 1	0 0 0 X	0 0

PIO instruction	Function	Modifier	Device address	
			Subaddress	Zeros
Digital output control (X = group)				
Immediate write	0 0 1			
Write digital output group register (X=group)		0 0 0 0	1 1 X X	0 0
Write digital output holding register		0 0 0 1	1 1 0 0	0 0
Set digital output group register		0 0 1 0	1 1 X X	0 0
Immediate read	0 1 0			
Read digital output group register (X=group)		0 0 0 0	1 1 X X	0 0
Read digital output holding register		0 0 0 1	1 1 0 0	0 0
Analog output control (X = point)				
Immediate write	0 0 1			
Write analog output register (X=point)		0 0 0 0	1 0 1 X	0 0
Write analog output holding register		0 0 0 1	1 0 1 0	0 0
Set analog output register		0 0 1 0	1 0 1 X	0 0
Immediate read	0 1 0			
Read analog output register		0 0 0 0	1 0 1 X	0 0
Read analog output holding register		0 0 0 1	1 0 1 0	0 0
2790 Control				
Immediate write	0 0 1			
Write address		0 0 0 0	1 0 0 0	0 0
Write control and data		0 0 0 1		
Set diagnostic mode		0 0 1 0		
Set input buffer control		0 0 1 1		
Immediate read	0 1 0			
Read address		0 0 0 0	1 0 0 0	0 0
Read control and data		0 0 0 1		
Read ISW		0 0 1 1		

PIO instruction	Function	Modifier	Device address	
			Subaddress	Zeros
Disk Storage				
Immediate write	0 0 1			
Write data		0 0 0 1	* * * *	* *
Load next sector ID		0 0 1 0		
Seek		0 0 1 1		
Format track		0 1 0 0		
Read verify		0 1 0 1		
Read initialize		0 1 1 0		
Write initialize		0 1 1 1		
Terminate		1 0 0 0		
Write diagnostic control		1 0 1 1		
Immediate read	0 1 0			
Read data		0 0 0 1	* * * *	* *
Read sector ID		0 0 1 0		
Read ISW		0 0 1 1		
Read ISWEX		1 0 1 1		
Read residual address		1 1 1 1		
Start read	1 1 0			
Read cycle steal		0 0 0 1	* * * *	* *
Read verify cycle steal		0 0 1 0		
Start write	1 1 1			
Write cycle steal		0 0 0 1	* * * *	* *
All I/O modules				
Immediate read	0 1 0			
Read module ID		0 1 0 0	* * * *	* *
Read module ID extension		0 1 0 1		
Read DSW		0 1 1 1		
Prepare I/O	0 1 1	* * * *	X X X X	0 0
Halt I/O	1 0 0	* * * *	X X X X	0 0
Processor module				
Immediate read	0 1 0			
Read direct control channel status word		* * * *	0 1 0 0	0 0
Set interrupt	1 0 1	* * * *	0 0 0 0	0 0

* * Indicates bits not used, but must be set to 0.

1130 I/O commands

Input/output control command (IOCC)	Device code	Function code	Modifier bits 8 9 10 11 12 13 14 15	Hexadecimal value
System/7 1130 host attachment	01100			
Initiate write				
No interrupt to System/7 for operation end		1 0 1	▒▒ 0 0 0 0	XXXX 6500
Interrupt System/7 for operation end		1 0 1	▒▒ 0 0 0 1	XXXX 6501
EIPL and no interrupt to System/7 for operation end		1 0 1	▒▒ 0 0 1 0	XXXX 6502
EIPL and interrupt System/7 for operation end		1 0 1	▒▒ 0 0 1 1	XXXX 6503
Initiate read				
No interrupt to System/7 for operation end		1 1 0	▒▒ 0 0 0 0	XXXX 6600
Interrupt System/7 for operation end		1 1 0	▒▒ 0 0 0 1	XXXX 6601
Control				
Transfer word count to System/7		1 0 0	▒▒ 0 0 1 0	XXXX 6402
Transfer System/7 storage address to System/7		1 0 0	▒▒ 0 0 1 1	XXXX 6403
Permit System/7 power /thermal warning interruptions		1 0 0	▒▒ 1 0 0 0	XXXX 6408
Prevent System/7 power/ thermal warning interruptions		1 0 0	▒▒ 1 1 0 0	XXXX 640C
Sense device				
Without reset		1 1 1	▒▒ 0 0 0 0	▒▒ 6700
With reset		1 1 1	▒▒ 0 0 0 1	▒▒ 6701

▒▒ Shaded areas are reserved. (The hexadecimal values were developed assuming all reserved modifiers to be zero.)

Valid responses to 2790 read commands

Read command	Read response						
	Request	Command ack	Data request	Data ack	End request	End ack	Null ack
Read		√					
Read Null	√		√		√		√
Read data				√			
Read end						√	

Valid responses to 2790 write commands

Write command	Write response					
	Data request	Data ack	End request	End ack	Null ack	Command ack
Write						√
Write null	√		√		√	
Write data		√				
Write end				√		

Valid responses to 2790 control commands

Control command	Control response
Bypass	Bypass acknowledge
Restore	Restore command repeated before data byte
Send Sync	Continuous stream of sync characters
Begin diagnostic	Begin diagnostic acknowledge
End diagnostic	End diagnostic acknowledge

Device status words

I/O Module	Device status word bits															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
5014 Analog Input Module		Command reject													Data check	
5012 Multifunction Module		Command reject		Digital input circuit breaker open											Data check	Invalid device address
5022 Disk Storage Module		Command reject			Inter-vention required										Data check	
5013 Digital I/O Module		Command reject		Digital input circuit breaker open											Data check	Invalid device address

Direct control channel status word

Status word bits															
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	Command reject		Operator station error	ACCA/ BSCA error											Invalid device address

Interrupt status words

Device	Interrupt status word bits															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
ACCA	Attention	Serial transmit data	Overrun	End of Block	Character received	Time-out	Ready to transmit	Data set error	Binary data mode		Receive mode		Device busy	Device end	Data check	
BSCA	Attention	Com-mand reject	Overrun	Time-out	No COD character	Diag-nostic mode	Data set ready	Data set error	Storage data check	Program check		Channel end	Device busy	Device end	Inter-face data check	Multi-point
Operator station	Attention			End of message									Device busy	Device end		Printer paper low
Analog input model B1				ADC inop		Multiple relays selected	No relays selected	Overload					Device busy	Device end		Invalid mpxr address
Analog input model C1				ADC inop		Invalid analog data		Overload					Device busy	Device end		Invalid mpxr address
Analog input model D1				ADC inop				Overload	Invalid range				Device busy	Device end		Invalid mpxr address
Digital input				Open fuse									Device busy	Device end		
2790				Frame parity error	No input detected	Diag mode	No sync bytes	Bypass Mode					Device busy	Device end		
Disk storage with DPC	Attention		Overrun	Any error	Drive became not ready	No record found		End of sector					Device busy	Device end		Read ISWEX
Disk Storage with cycle steal	Attention		Overrun	Any error	Drive became not ready	No record found	Incorrect length record		Storage data check	Program check		Channel end	Device busy	Device end	Inter-face data check	Read ISWEX

1130 device status word

Device	Device status word bits															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
System/7	Attention	Operation end	Invalid storage address	Data check	Count equals zero	Power/ thermal warning	Storage control check								Ready	Busy

Interrupt status word extension

Device		Interrupt status word extension bits															
		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Disk storage module	Normal position	File data unsafe	Missing address mark	Equip-ment check	Seek check	File data check	File sense 1	File sense 2	File sense 3		Parallel parity check	Check counter error	Write gate check		Cylinder zero	Diag. mode	Write inhibited
	Diagnostic position	File data unsafe	Missing address mark	Equip-ment check	Seek check	File data check	Head settling time	Address mark time	Pre-ID time	Bit ring zero	Write data to file	CC register pos. 17	Bit ring inhibit	Index time 4	Data buffer busy	Index sensed	IPL select lower

Processor status word

Device	Processor status word bits															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Processor module	Invalid shift count	Invalid storage address	Invalid Op code funct. or mod.	Local storage parity check	SDR parity check	Control check	I/O check	Sequence indicator	Power warning (master unit)	Thermal warning (master unit)	Power warning (remote unit)	Thermal warning (remote unit)				

Device status word (2502, 5024 with Printer)

I/O Module	Device status word bits															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
2502 card reader word 1	Any error	EBCDIC checksum	Command reject	Reader not ready	Equipment check	Data check	End of file	EOF switch	Storage data check	Program check	Hardware checksum error	Invalid command	2502 attached	Device end	Printer attached	Status word 2 indicator
2502 card reader word 2	Read emitter sync check	Feed check	Read check	Stacker full, jam	Read emitter sync check	Trailing edge sync check	FCB2 error	FCB1 error	Cover open	Hardware error	No holes check	Read skew check	Zeros			
5024 with printer word 1	Any error	End of forms	Command reject	Not ready	Form overflow	Invalid belt image	Incorrect record length	Belt speed check	Storage data check	Program check	Hardware checksum error	Invalid command	2502 attached	Device end	Printer attached	Status word 2 indicator
5024 with printer word 2	Throat open	End of forms	Cover open	ENABLE/ DISABLE switch in DISABLE	Overtemperature	Forms check light on	Binary "0"	Binary "0"	Printer check light	Hammer check	Binary "0"	Coded error status (hardware error)				

Appendix D. Character Codes

Dec	Hex	Binary	BSCA EBCDIC	BSCA ASCII	5028 Console Shift/ Control (Note 1)	ACCA PTTC/EBCD	5028 Paper Tape	80-column Card Code	96-column Card Code
0	0 0	0000 0000	NUL	NUL	(P) SC	:	blank	12-0-1-8-9	blank
1	0 1	0001	SOH			space	C	12-1-9	BA1
2	0 2	0010	STX			1	1	12-2-9	BA2
3	0 3	0011	ETX	ETX	C		1C	12-3-9	BA21
4	0 4	0100	PF			2	2	12-4-9	BA4
5	0 5	0101	HT	ENQ	(E) C		2C	12-5-9	BA41
6	0 6	0110	LC	ACK	(F) C		21	12-6-9	BA42
7	0 7	0111	DEL			3	21C	12-7-9	BA421
8	0 8	0000 1000				4	4	12-8-9	BA8
9	0 9	1001		TAB	C		4C	12-1-8-9	BA81
10	0 A	1010	SMM	LF	(J) C		41	12-2-8-9	BA82
11	0 B	1011	VT			5	41C	12-3-8-9	BA821
12	0 C	1100	FF	FF	C		42	12-4-8-9	BA84
13	0 D	1101	CR			6	42C	12-5-8-9	BA841
14	0 E	1110	SO			7	421	12-6-8-9	BA842
15	0 F	1111	SI	SI	(O) C		421C	12-7-8-9	BA8421
16	1 0	0001 0000	DLE			8	8	12-11-1-8-9	A82
17	1 1	0001	DC1	DC1	C		8C	11-1-9	B1
18	1 2	0010	DC2	DC2	C		81	11-2-9	B2
19	1 3	0011	TM (DC3)			9	81C	11-3-9	B21
20	1 4	0100	RES	DC4	C		82	11-4-9	B4
21	1 5	0101	NL			0	82C	11-5-9	B41
22	1 6	0110	BS			D,EOA,#	821	11-6-9	B42
23	1 7	0111	IL	ETB	C		821C	11-7-9	B421
24	1 8	0001 1000	CAN	CAN	C		84	11-8-9	B8
25	1 9	1001	EM				84C	11-1-8-9	B81
26	1 A	1010	CC				841	11-2-8-9	B82
27	1 B	1011	CU1	ESC			841C	11-3-8-9	B821
28	1 C	1100	IFS			up-shift	842	11-4-8-9	B84
29	1 D	1101	IGS	GS	(M) SC		842C	11-5-8-9	B841
30	1 E	1110	IRS	RS	(N) SC		8421	11-6-8-9	B842
31	1 F	1111	IUS			C,EOT	8421C	11-7-8-9	B8421
32	2 0	0010 0000	DS			@	A	11-0-1-8-9	B
33	2 1	0001	SOS	!	S		AC	0-1-9	AI
34	2 2	0010	FS	"	S		A1	0-2-9	A2
35	2 3	0011				/	A1C	0-3-9	A21
36	2 4	0100	BYP	$	S		A2	0-4-9	A4
37	2 5	0101	LF			s	A2C	0-5-9	A41
38	2 6	0110	ETB			t	A21	0-6-9	A42
39	2 7	0111	ESC	Appos	S		A21C	0-7-9	A421
40	2 8	0010 1000		(	S		A4	0-8-9	A8
41	2 9	1001				u	A4C	0-1-8-9	A81
42	2 A	1010	SM			v	A41	0-2-8-9	BA
43	2 B	1011	CU2	+	S		A41C	0-3-8-9	A821
44	2 C	1100				w	A42	0-4-8-9	A84
45	2 D	1101	ENQ	–			A42C	0-5-8-9	A841
46	2 E	1110	ACK	period			A421	0-6-8-9	A842
47	2 F	1111	BEL			x	A421C	0-7-8-9	A8421

Dec	Hex	Binary	BSCA EBCDIC	BSCA ASCII	5028 Console Shift/ Control (Note 1)	ACCA PTTC/EBCD	5028 Paper Tape	80-column Card Code	96-column Card Code
48	3 0	0011 0000		0			A8	12-11-0-1-8-9	A
49	3 1	0001				y	A8C	1-9	1
50	3 2	0010	SYN			z	A81	2-9	2
51	3 3	0011		3			A81C	3-9	21
52	3 4	0100	PN				A82	4-9	4
53	3 5	0101	RS	5			A82C	5-9	41
54	3 6	0110	UC	6			A821	6-9	42
55	3 7	0111	EOT			S,SOA, comma	A821C	7-9	421
56	3 8	0011 1000					A84	8-9	8
57	3 9	1001		9			A84C	1-8-9	81
58	3 A	1010		:			A841	2-8-9	82
59	3 B	1011	CU3			LF (see note 2)	A841C	3-8-9	821
60	3 C	1100	DC4	<	S		A842	4-8-9	84
61	3 D	1101	NAK			B,EOB	A842C	5-8-9	841
62	3 E	1110					A8421	6-8-9	842
63	3 F	1111	SUB	?	S		A8421C	7-8-9	8421
64	4 0	0100 0000	SP			N,–	B	blank	blank
65	4 1	0001		A			BC	12-0-1-9	BA1
66	4 2	0010		B			B1	12-0-2-9	BA2
67	4 3	0011				j	B1C	12-0-3-9	BA21
68	4 4	0100		D			B1	12-0-4-9	BA4
69	4 5	0101				k	B2C	12-0-5-9	BA41
70	4 6	0110				l	B21	12-0-6-9	BA42
71	4 7	0111		G			B21C	12-0-7-9	BA421
72	4 8	0100 1000		H			B4	12-0-8-9	BA8
73	4 9	1001				m	B4C	12-1-8	BA81
74	4 A	1010	¢			n	B41	12-2-8	BA82
75	4 B	1011	period	K			B41C	12-3-8	BA821
76	4 C	1100	<			o	B42	12-4-8	BA84
77	4 D	1101	(	M			B42C	12-5-8	BA841
78	4 E	1110	+	N			B421	12-6-8	BA842
79	4 F	1111	]			p	B421C	12-7-8	BA8421
80	5 0	0101 0000	&	P			B8	12	A82
81	5 1	0001				q	B8C	12-11-1-9	B1
82	5 2	0010				r	B81	12-11-2-9	B2
83	5 3	0011		S			B81C	12-11-3-9	B21
84	5 4	0100					B82	12-11-4-9	B4
85	5 5	0101		U			B82C	12-11-5-9	B41
86	5 6	0110		V			B821	12-11-6-9	B42
87	5 7	0111				$	B821C	12-11-7-9	B421
88	5 8	0101 1000					B84	12-11-8-9	B8
89	5 9	1001		Y			B84C	11-1-8	B81
90	5 A	1010	!	Z			B841	11-2-8	B82
91	5 B	1011	$			NL (see note 3)	B841C	11-3-8	B821
92	5 C	1100	*		S		B842	11-4-8	B84
93	5 D	1101	)			back-space	B842C	11-5-8	B841
94	5 E	1110	;			IDLE	B8421	11-6-8	B842
95	5 F	1111	¬	under-score	S		B8421C	11-7-8	B8421

Dec	Hex	Binary	BSCA EBCDIC	BSCA ASCII	5028 Console Shift/ Control (Note 1)	ACCA PTTC/EBCD	5028 Paper Tape	80-column Card Code	96-column Card Code
96	6 0	0110 0000	–				BA	11	B
97	6 1	0001	/			&	BAC	0-1	A1
98	6 2	0010				a	BA1	11-0-2-9	A2
99	6 3	0011					BA1C	11-0-3-9	A21
100	6 4	0100				b	BA2	11-0-4-9	A4
101	6 5	0101					BA2C	11-0-5-9	A41
102	6 6	0110					BA21	11-0-6-9	A42
103	6 7	0111				c	BA21C	11-0-7-9	A421
104	6 8	0110 1000				d	BA4	11-0-8-9	A8
105	6 9	1001					BA4C	0-1-8	A81
106	6 A	1010	EOM				BA41	12-11	BA82
107	6 B	1011	,			e	BA41C	0-3-8	A821
108	6 C	1100	%				BA42	0-4-8	A84
109	6 D	1101	underscore			f	BA42C	0-5-8	A841
110	6 E	1110	>			g	BA421	0-6-8	A842
111	6 F	1111	?				BA421C	0-7-8	A8421
112	7 0	0111 0000				h	BA8	12-11-0	A
113	7 1	0001					BA8C	12-11-0-1-9	1
114	7 2	0010					BA81	12-11-0-2-9	2
115	7 3	0011				i	BA81C	12-11-0-3-9	21
116	7 4	0100					BA82	12-11-0-4-9	4
117	7 5	0101					BA82C	12-11-0-5-9	41
118	7 6	0110				Y,period	BA821	12-11-0-6-9	42
119	7 7	0111					BA821C	12-11-0-7-9	421
120	7 8	0111 1000					BA84	12-11-0-8-9	8
121	7 9	1001					BA84C	1-8	81
122	7 A	1010	:			horiz. tab	BA841	2-8	82
123	7 B	1011	#				BA841C	3-8	821
124	7 C	1100	@			down-shift	BA842	4-8	84
125	7 D	1101	'				BA842C	5-8	841
126	7 E	1110	=				BA8421	6-8	842
127	7 F	1111	"			DEL	BA8421C	7-8	8421
128	8 0	1000 0000					S	12-0-1-8	blank
129	8 1	0001	a	SOH	C	space	SC	12-0-1	BA1
130	8 2	0010	b	STX	C	=	S1	12-0-2	BA2
131	8 3	0011	c				S1C	12-0-3	BA21
132	8 4	0100	d	EOT	C	<	S2	12-0-4	BA4
133	8 5	0101	e				S2C	12-0-5	BA41
134	8 6	0110	f				S21	12-0-6	BA42
135	8 7	0111	g	BELL	C	;	S21C	12-0-7	BA421
136	8 8	1000 1000	h	BS	(H) C	:	S4	12-0-8	BA8
137	8 9	1001	i				S4C	12-0-9	BA81
138	8 A	1010					S41	12-0-2-8	BA82
139	8 B	1011		VT	C	%	S41C	12-0-3-8	BA821
140	8 C	1100					S42	12-0-4-8	BA84
141	8 D	1101		CR	(M) C	'	S42C	12-0-5-8	BA841
142	8 E	1110		SO	(N) C	>	S421	12-0-6-8	BA842
143	8 F	1111					S421C	12-0-7-8	BA8421

Dec	Hex	Binary	BSCA EBCDIC	BSCA ASCII	5028 Console Shift/ Control (Note 1)	ACCA PTTC/EBCD	5028 Paper Tape	80-column Card Code	96-column Card Code
144	9 0	1001 0000		DLE	(P) C	*	S8	12-11-1-8	BA
145	9 1	0001	j				S8C	12-11-1	B1
146	9 2	0010	k				S81	12-11-2	B2
147	9 3	0011	l	DC3	C	(	S81C	12-11-3	B21
148	9 4	0100	m				S82	12-11-4	B4
149	9 5	0101	n	NAK	(U) C	)	S82C	12-11-5	B41
150	9 6	0110	o	SYN	C		S821	12-11-6	B42
151	9 7	0111	p				S821C	12-11-7	B421
152	9 8	1001 1000	q				S84	12-11-8	B8
153	9 9	1001	r	EM	C		S84C	12-11-9	B1
154	9 A	1010		SUB	C		S841	12-11-2-8	B82
155	9 B	1011					S841C	12-11-3-8	B821
156	9 C	1100		FS	(L) CS	up-shift	S842	12-11-4-8	B84
157	9 D	1101					S842C	12-11-5-8	B841
158	9 E	1110					S8421	12-11-6-8	B842
159	9 F	1111		US	(O) CS		S8421C	12-11-7-8	B8421
160	A 0	1010 0000		space		¢	SA	11-0-1-8	B
161	A 1	0001					SAC	11-0-1	A1
162	A 2	0010	s				SA1	11-0-2	A2
163	A 3	0011	t	#	S	?	SA1C	11-0-3	A21
164	A 4	0100	i				SA2	11-0-4	A4
165	A 5	0101	v	%	S	S	SA2C	11-0-5	A41
166	A 6	0110	w	&	S	T	SA21	11-0-6	A42
167	A 7	0111	x				SA21C	11-0-7	A421
168	A 8	1010 1000	y				SA4	11-0-8	A8
169	A 9	1001	z	)	S	U	SA4C	11-0-9	A81
170	A A	1010		*	S	V	SA41	11-0-2-8	A82
171	A B	1011					SA41C	11-0-3-8	A821
172	A C	1100		comma		W	SA42	11-0-4-8	A84
173	A D	1101					SA42C	11-0-5-8	A841
174	A E	1110					SA421	11-0-6-8	A842
175	A F	1111		/		X	SA421C	11-0-7-8	A8421
176	B 0	1011 0000					SA8	12-11-0-1-8	A
177	B 1	0001		1		Y	SA8C	12-11-0-1	1
178	B 2	0010		2		Z	SA81	12-11-0-2	2
179	B 3	0011					SA81C	12-11-0-3	21
180	B 4	0100		4			SA82	12-11-0-4	4
181	B 5	0101					SA82C	12-11-0-5	41
182	B 6	0110					SA821	12-11-0-6	42
183	B 7	0111		7			SA821C	12-11-0-7	421
184	B 8	1011 1000		8			SA84	12-11-0-8	8
185	B 9	1001					SA84C	12-11-0-9	81
186	B A	1010					SA841	12-11-0-2-8	82
187	B B	1011		;		LF (note 2)	SA841C	12-11-0-3-8	821
188	B C	1100					SA842	12-11-0-4-8	84
189	B D	1101		=	S	B,EOB	SA842C	12-11-0-5-8	841
190	B E	1110		>	S		SA8421	12-11-0-6-8	842
191	B F	1111					SA8421C	12-11-0-7-8	8421

Dec	Hex	Binary	BSCA EBCDIC	BSCA ASCII	5028 Console Shift/ Control (Note 1)	ACCA PTTC/EBCD	5028 Paper Tape	80-column Card Code	96-column Card Code
192	C 0	1100 0000		@	S	N	SB	12-0	blank
193	C 1	0001	A				SBC	12-1	BA1
194	C 2	0010	B				SB1	12-2	BA2
195	C 3	0011	C	C		J	SB1C	12-3	BA21
196	C 4	0100	D				SB2	12-4	BA4
197	C 5	0101	E	E		K	SB2C	12-5	BA41
198	C 6	0110	F	F		L	SB21	12-6	BA42
199	C 7	0111	G				SB21C	12-7	BA421
200	C 8	1100 1000	H				SB4	12-8	BA8
201	C 9	1001	I	I		M	SB4C	12-9	BA81
202	C A	1010		J		N	SB41	12-0-2-8-9	BA82
203	C B	1011					SB41C	12-0-3-8-9	BA821
204	C C	1100		L		O	SB42	12-0-4-8-9	BA84
205	C D	1101					SB42C	12-0-5-8-9	BA841
206	C E	1110					SB421	12-0-6-8-9	BA842
207	C F	1111		O		P	SB421C	12-0-7-8-9	BA8421
208	D 0	1101 0000					SBB	11-0	BA
209	D 1	0001	J	Q		Q	SB8C	11-1	B1
210	D 2	0010	K	R		R	SB81	11-2	B2
211	D 3	0011	L				SB81C	11-3	B21
212	D 4	0100	M	T			SB82	11-4	B4
213	D 5	0101	N				SB82C	11-5	B41
214	D 6	0110	O				SB821	11-6	B42
215	D 7	0111	P	W		!	SB821C	11-7	B421
216	D 8	1101 1000	Q	X			SB84	11-8	B8
217	D 9	1001	R				SB84C	11-9	B81
218	D A	1010					SB841	12-11-2-8-9	B82
219	D B	1011		[	S	NL (note 3)	SB841C	12-11-3-8-9	B82
220	D C	1100					SB842	12-11-4-8-9	B84
221	D D	1101		]	S	back-space	SB842C	12-11-5-8-9	B841
222	D E	1110				IDLE	SB8421	12-11-6-8-9	B842
223	D F	1111					SB8421C	12-11-7-8-9	B8421
224	E 0	1110 0000					SBA	0-2-8	B
225	E 1	0001				+	SBAC	11-0-1-9	A1
226	E 2	0010	S			A	SBA1	0-2	A2
227	E 3	0011	T				SBA1C	0-3	A21
228	E 4	0100	U			B	SBA2	0-4	A4
229	E 5	0101	V				SBA2C	0-5	A41
230	E 6	0110	W				SBA21	0-6	A42
231	E 7	0111	X			C	SBA21C	0-7	A421
232	E 8	1110 1000	Y			D	SBA4	0-8	A8
233	E 9	1001	Z				SBA4C	0-9	A81
234	E A	1010					SBA41	11-0-2-8-9	A82
235	E B	1011				E	SBA41C	11-0-3-8-9	A821
236	E C	1100					SBA42	11-0-4-8-9	A84
237	E D	1101				F	SBA42C	11-0-5-8-9	A841
238	E E	1110				G	SBA421	11-0-6-8-9	A842
239	E F	1111					SBA421C	11-0-7-8-9	A8421

Dec	Hex	Binary	BSCA EBCDIC	BSCA ASCII	5028 Console Shift/ Control (Note 1)	ACCA PTTC/EBCD	5028 Paper Tape	80-column Card Code	96-column Card Code
240	F 0	1111 0000	0			H	SBA8	0	A
241	F 1	0001	1				SBA8C	1	1
242	F 2	0010	2				SBA81	2	2
243	F 3	0011	3			I	SBA81C	3	21
244	F4	0100	4				SBA82	4	4
245	F 5	0101	5				SBA82C	5	41
246	F 6	0110	6			Y	SBA821	6	42
247	F 7	0111	7				SBA821C	7	421
248	F 8	1111 1000	8				SBA84	8	8
249	F 9	1001	9				SBA84C	9	81
250	F A	1010				horiz. tab	SBA841	12-11-0-2-8-9	82
251	F B	1011					SBA841C	12-11-0-3-8-9	821
252	F C	1100				down-shift	SBA842	12-11-0-4-8-9	84
253	F D	1101					SBA842C	12-11-0-5-8-9	841
254	F E	1110					SBA8421	12-11-0-6-8-9	842
255	F F	1111		DEL		DEL	SBA8421C	12-11-0-7-8-9	8421

Note 1. The shift/control column indicates the combination of one, two or three 5028 keyboard keys that will produce the equivalent code when that code is not valid for the 5028 printer. If the graphic character or control name does not appear on the keyboard, the character key (to be used) is enclosed within parentheses in this column. The letter C, in this column, specifies that the control key must be used with the character key. The letter S, in this column, specifies that the shift key must be used with the character key.

Note 2. LF (line feed) is used to index.

Note 3. NL (new line) is used for a carrier return and line feed.

CONVERTING ANALOG INPUT BINARY DATA TO VOLTAGE

The input voltage converted by the analog-to-digital converter is represented in a 16-bit data word by either 12 or 14 binary bits plus a sign bit, as shown in Figure E-1, depending on the amplifier range selected and the read command used. The voltage value that each binary bit represents for a particular range is called the resolution for that range. The resolution for each analog input range is shown in Figure E-2.

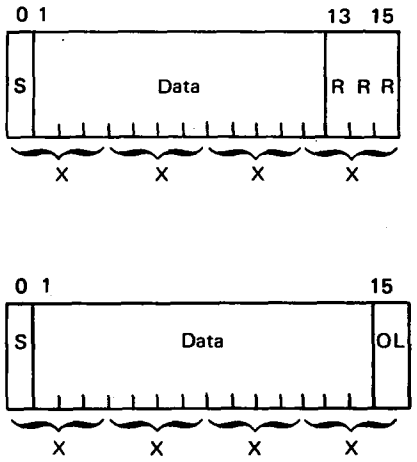


Figure E-1. Analog input data words

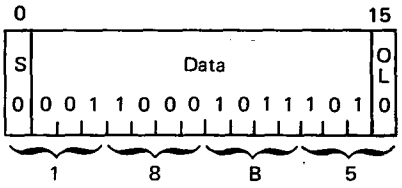
Figure E-2. Analog input resolution

Bit	Ranges and bit values (in millivolts)						
	10 millivolt	20 millivolt	40 millivolt	80 millivolt	160 millivolt	640 millivolt	5.12 volt
1	5.0000000	10.0000000	20.0000000	40.0000000	80.0000000	320.0000000	2560.0000
2	2.5000000	5.0000000	10.0000000	20.0000000	40.0000000	160.0000000	1280.0000
3	1.2500000	2.5000000	5.0000000	10.0000000	20.0000000	80.0000000	640.0000
4	0.6250000	1.2500000	2.5000000	5.0000000	10.0000000	40.0000000	320.0000
5	0.3125000	0.6250000	1.2500000	2.5000000	5.0000000	20.0000000	160.0000
6	0.1562500	0.3125000	0.6250000	1.2500000	2.5000000	10.0000000	80.0000
7	0.0781250	0.1562500	0.3125000	0.6250000	1.2500000	5.0000000	40.0000
8	0.0390625	0.0781250	0.1562500	0.3125000	0.6250000	2.5000000	20.0000
9	0.0195312	0.0390625	0.0781250	0.1562500	0.3125000	1.2500000	10.0000
10	0.0097656	0.0195312	0.0390625	0.0781250	0.1562500	0.6250000	5.0000
11	0.0048828	0.0097656	0.0195312	0.0390625	0.0781250	0.3125000	2.5000
12	0.0024414	0.0048828	0.0097656	0.0195312	0.0390625	0.1562500	1.2500
13	0.0012207	0.0024414	0.0048828	0.0097656	0.0195312	0.0781250	0.6250
14	0.0006103	0.0012207	0.0024414	0.0048828	0.0097656	0.0390625	0.3125

Note: Multiply the above values by 10^3 to obtain the equivalent voltage in microvolts. To obtain the equivalent voltage in volts, multiply the above values by 10^{-3}

Conversion of a data word into its corresponding voltage is made by adding the voltage values represented by the binary bits that are set on.

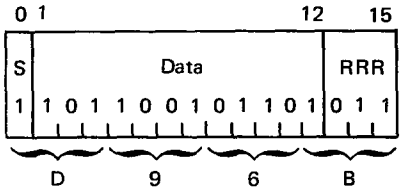
Example 1: Amplifier range selected—160 millivolts. Data word from ADC:



The conversion is made from Figure E-2 as follows:

Bit Set	*Value (millivolts)*
3	20.0000000
4	10.0000000
8	0.6250000
10	0.1562500
11	0.0781250
12	0.0390625
14	0.0097656
Total	+30.9082031 millivolts

Example 2: Amplifier range selected—automatic. Data word from ADC:



The range bits (RRR)=011 specify the 40 millivolt range. The twos complement of the data word above must be used for the conversion, which is made from Figure E-2 as follows:

Complement Bit	*Value (millivolts)*
2	10.0000000
5	1.2500000
6	0.6250000
8	0.1562500
11	0.0195312
12	0.0097656
Total	−12.0605468 millivolts

CONVERTING VOLTAGE TO ANALOG OUTPUT BINARY DATA

The output voltage signal to be established at an analog output point is represented by 10 binary bits in a 16-bit data word as shown in Figure E-3.

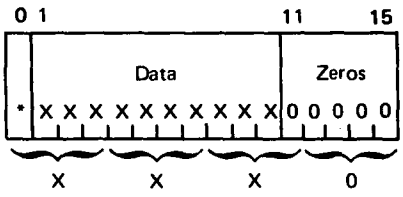


Figure E-3. Analog output data word

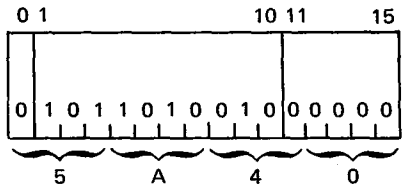
The voltage value represented by each binary bit is shown in Figure E-4. Converting a voltage to its binary equivalent can be accomplished by using the flowchart shown in Figure E-5.

	Bits											
	0	1	2	3	4	5	6	7	8	9	10	11 through 15
Voltage value →	Not used	5.12	2.56	1.28	0.640	0.320	0.160	0.080	0.040	0.020	0.010	Not used

Note: All voltage values shown are expressed in volts.

Figure E-4. Analog output bit resolution

For example, converting the voltage 7.225 using the flowchart would result in the following data word:



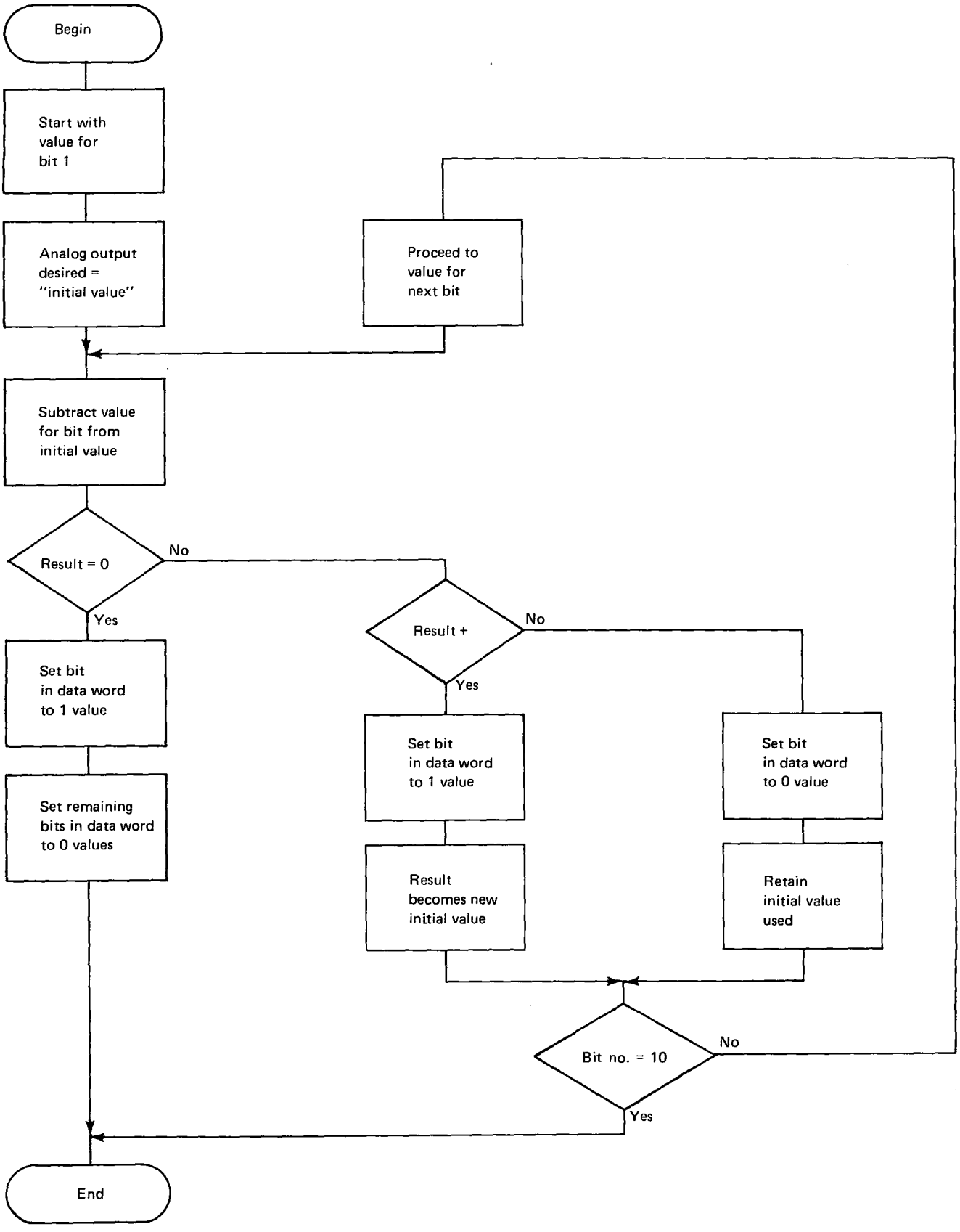


Figure E-5. Analog output conversion flowchart

Note. In all charts of this appendix, the I/O commands are enclosed in bold-lined boxes.

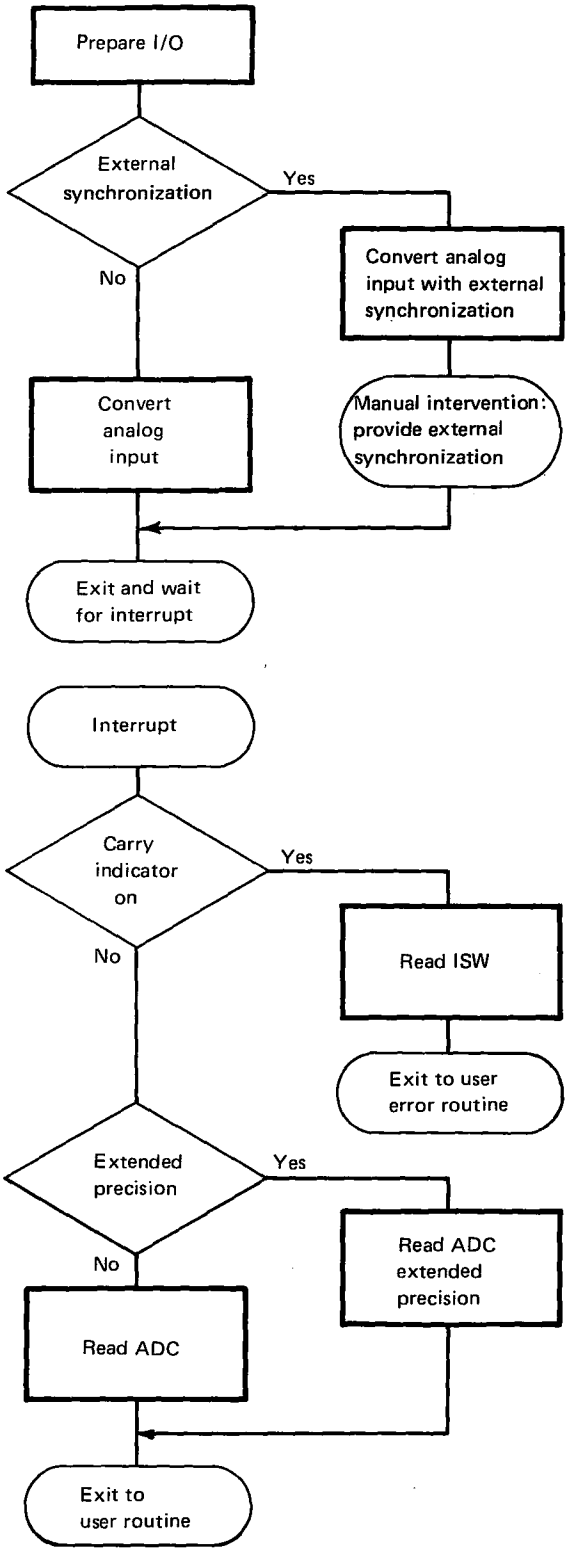


Figure F-1. Analog input routine

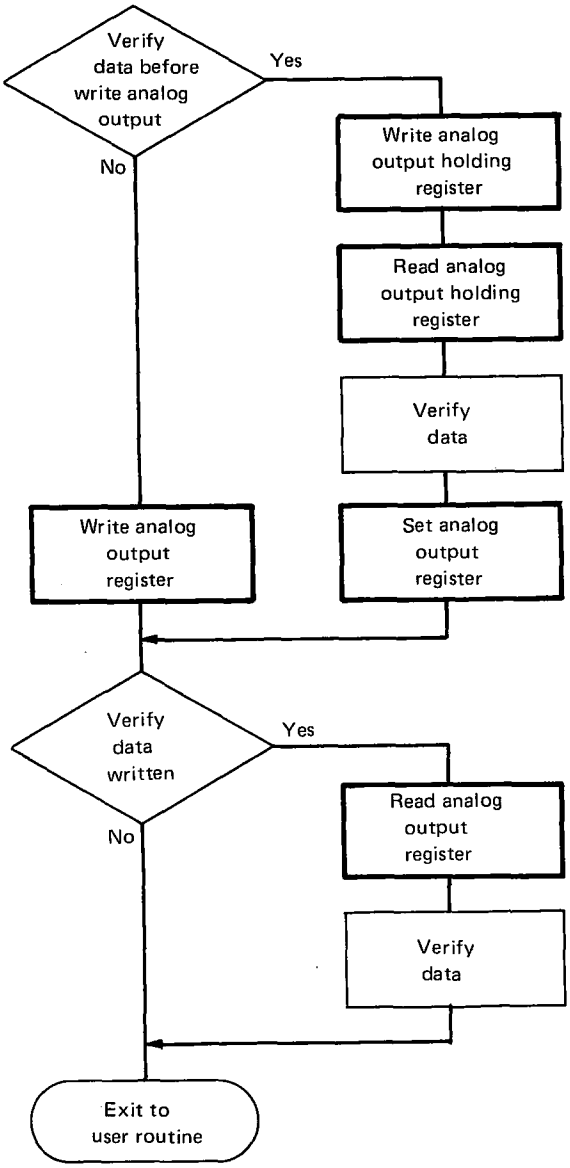


Figure F-2. Analog output routine

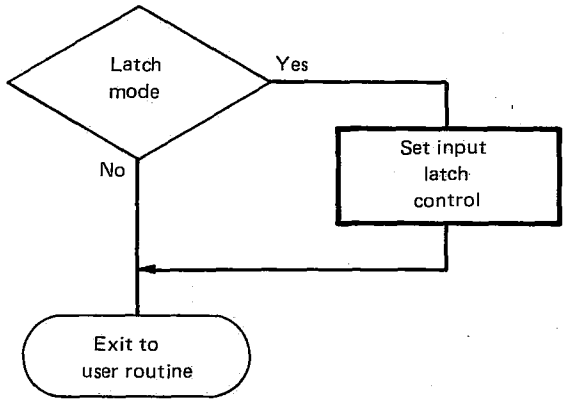


Figure F-3. Digital input routine without the Process Interrupt feature

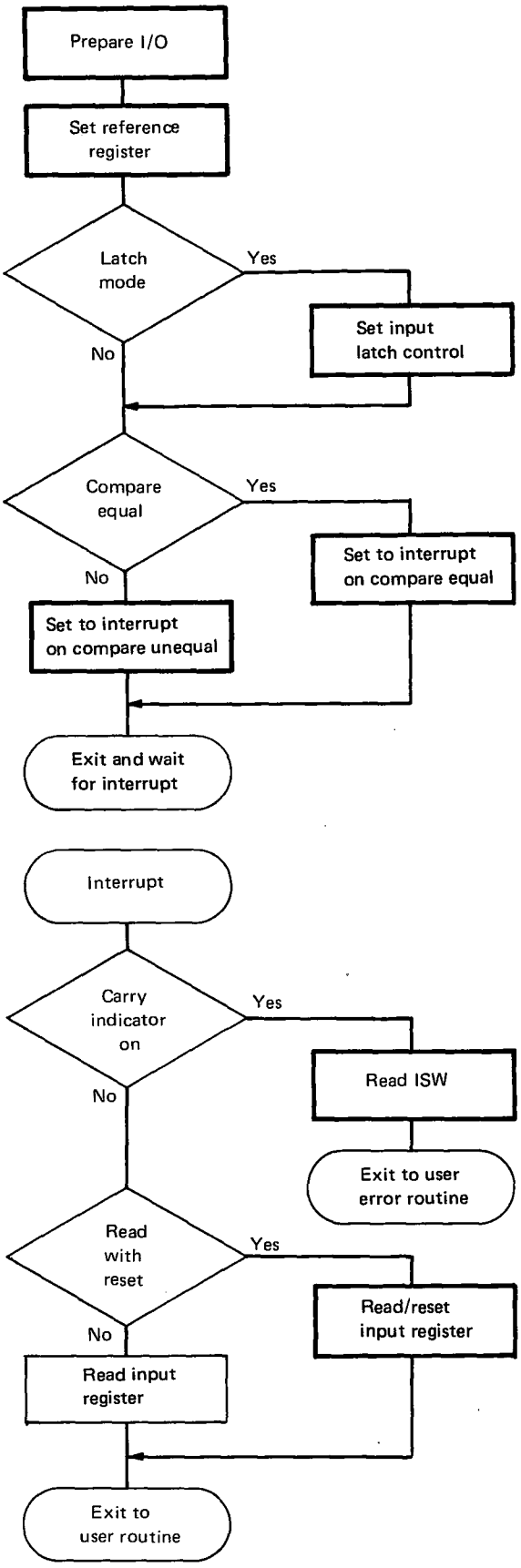


Figure F-4. Digital input routine with the Process Interrupt feature

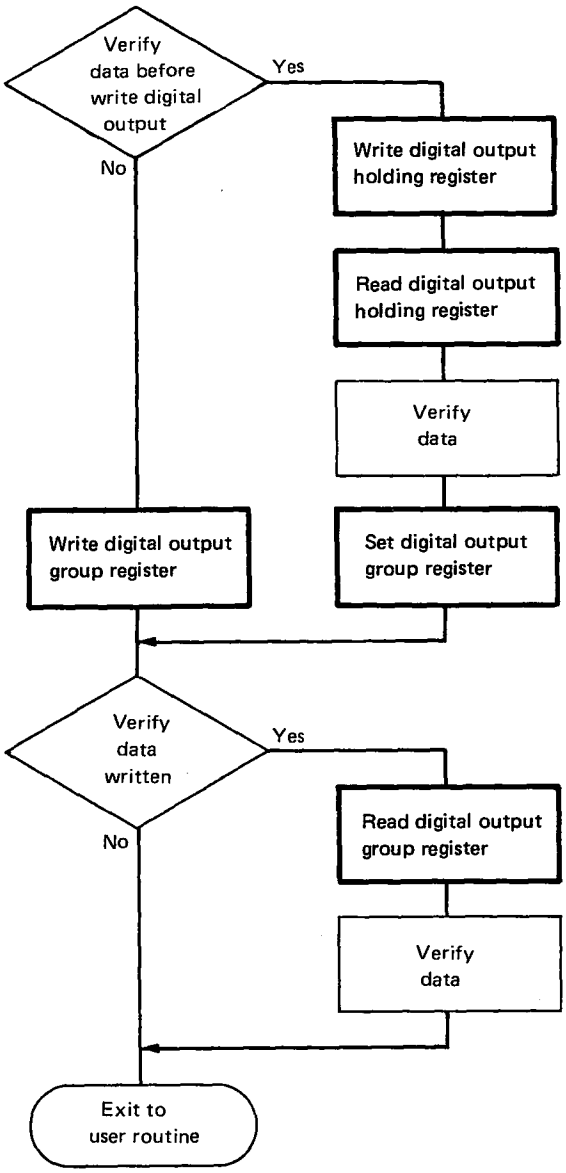


Figure F-5. Digital output routine

Appendix G. Maximum Sampling Rates of Sensor-Based I/O

IBM System/7 Installation Manual-Physical Planning, GA34-0004 provides detailed information about sampling rates of the various types of sensor-based I/O. As there are infinite combinations of loads, voltages, currents, and line length which effect sampling rates, the figures presented here are *maximum* and for reference only. GA34-0004 should be consulted for specific applications.

I/O type	Description of application	Max bits/second
DI	Voltage sense (isolated) for logic level signals, short distance, high speed.	300,000
	Voltage sense (non-isolated) for logic level signals, short distance, high speed.	185,000
	Voltage sense for logic level signals, medium distance, medium speed.	10,000
	Voltage sense for high level signals, medium distance, medium speed.	10,000
	Voltage sense for logic level signals, long distance, high speed.	300,000
	Voltage sense for high level signals, long distance, high speed.	300,000
	Contact sense.	250
		Response time
DO	Low power group	5.8 μsec
	Medium power group (non-isolated)	3.7 μsec
	Medium power group (isolated)	5.8 μsec
	Contact group	4 msec
		Settling time
AO		40 μsec
		Max points/second
AI	Models B1 and E2	200
	Model C1 (high level inputs)	20,000
	Model C1 (low level inputs)	14,000
	Model C1 (automatic gain selection)	7,000
	Models D1 and E1	100

IBM System/7
Functional Characteristics
GA34-0003-7

READER'S COMMENT FORM

Your views about this publication may help improve its usefulness; this form will be sent to the author's department for appropriate action. Using this form to request system assistance or additional publications will delay response, however. *For more direct handling of such requests, please contact your IBM representative or the IBM Branch Office serving your locality.*

Possible topics for comment are:

Clarity Accuracy Completeness Organization Index Figures Examples Legibility

What is your occupation? _____

Number of latest Technical Newsletter (if any) concerning this publication: _____

Please indicate your name and address in the space below if you wish a reply.

_____

_____

_____

Thank you for your cooperation. No postage stamp necessary if mailed in the U.S.A. (Elsewhere, an IBM office or representative will be happy to forward your comments.)

Cut or Fold Along Line

Your comments, please . . .

This manual is part of a library that serves as a reference source for systems analysts, programmers, and operators of IBM systems. Your comments on the other side of this form will be carefully reviewed by the persons responsible for writing and publishing this material. All comments and suggestions become the property of IBM.

Fold Fold

First Class
Permit 40
Armonk
New York

Business Reply Mail
No postage stamp necessary if mailed in the U.S.A.

IBM Corporation
Systems Publications, Dept 27T
P.O. Box 1328
Boca Raton, Florida 33432

Fold Fold

IBM

International Business Machines Corporation
General Systems Division
5775D Glenridge Drive, Atlanta, Georgia 30328
(U.S.A. only)

IBM World Trade Corporation
821 United Nations Plaza, New York, New York 10017
(International)

Cut Along Line

IBM System/7 Functional Characteristics (File No. S7-01) Printed in U.S.A. GA34-0003-7

IBM Technical Newsletter

This Newsletter No. GN34-0340
Date August 20, 1976
Base Publication No. GA34-0003-7
File No. S7-00
Previous Newsletters GN34-0290 (-6 only)

IBM System/7
Functional Characteristics

This Technical Newsletter provides replacement pages for the subject publication. Pages to be inserted and/or removed are:

iii, iv
2-11, 2-12
3-5 through 3-8
3-11, 3-12
4-61, 4-62
8-5 through 8-8
8-11, 8-12
13-37, 13-38
14-23, 14-24

A technical change to the text or to an illustration is indicated by a vertical line to the left of the change.

Summary of Amendments

The BSCA I/O Commands and Device Control Block description in chapter 8 is changed. In chapters 13 and 14, the 2790 status characters have been amended. Typographical errors have been corrected in chapters 2, 3, and 4.

Note. Please file this cover letter at the back of the manual to provide a record of changes.

IBM Corporation, Systems Publications, Department 27T, P. O. Box 1328, Boca Raton, Florida 33432

Printed in U.S.A.

IBM System/7
Functional Characteristics
GA34-0003-7

READER'S COMMENT FORM

Your views about this publication may help improve its usefulness; this form will be sent to the author's department for appropriate action. Using this form to request system assistance or additional publications will delay response, however. *For more direct handling of such requests, please contact your IBM representative or the IBM Branch Office serving your locality.*

Possible topics for comment are:

Clarity Accuracy Completeness Organization Index Figures Examples Legibility

What is your occupation? _____

Number of latest Technical Newsletter (if any) concerning this publication: _____

Please indicate your name and address in the space below if you wish a reply.

_____

_____

_____

Thank you for your cooperation. No postage stamp necessary if mailed in the U.S.A. (Elsewhere, an IBM office or representative will be happy to forward your comments.)

Cut or Fold Along Line

Your comments, please . . .

This manual is part of a library that serves as a reference source for systems analysts, programmers, and operators of IBM systems. Your comments on the other side of this form will be carefully reviewed by the persons responsible for writing and publishing this material. All comments and suggestions become the property of IBM.

Fold Fold

First Class
Permit 40
Armonk
New York

Business Reply Mail
No postage stamp necessary if mailed in the U.S.A.

IBM Corporation
Systems Publications, Dept 27T
P.O. Box 1328
Boca Raton, Florida 33432

Fold Fold

Cut Along Line

IBM System/7 Functional Characteristics (File No. S7-01) Printed in U.S.A. GA34-0003-7

IBM

International Business Machines Corporation
General Systems Division
5775D Glenridge Drive, Atlanta, Georgia 30328
(U.S.A. only)

IBM World Trade Corporation
821 United Nations Plaza, New York, New York 10017
(International)

GA34-0003-7

IBM System/7 Functional Characteristics (File No. S7-01) Printed in U.S.A. GA34-0003-7

IBM

International Business Machines Corporation
General Systems Division
5775D Glenridge Drive, Atlanta, Georgia 30328
(U.S.A. only)

IBM World Trade Corporation
821 United Nations Plaza, New York, New York 10017
(International)