IBM System/3
Models 8, 10, 12, and 15
Components
Reference Manual

GA21-9236-1
File No. S3-01

Second Edition (November 1977)

This is a major revision of, and obsoletes, GA21-9236-0 and Technical Newsletters GN21-0257, GN21-0262, and GN21-0270. Information about Binary Synchronous Communications Controller (BSCC) and IBM System/3 Model 15 D25 (384K memory) and Model 15 D26 (512K memory) have been added. Because the changes and additions are extensive, this manual should be reviewed in its entirety.

Changes are periodically made to the information herein. Before using this publication in connection with the operation of IBM systems, refer to the latest *IBM System/3 Bibliography*, GC20-8080, for the editions that are applicable and current.

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

Address comments concerning the contents of this publication to IBM Corporation, Publications, Department 245, Rochester, Minnesota 55901. Comments become the property of IBM.

Preface

This publication is intended for computer programmers, systems analysts, engineers, and others interested in machine language or in basic assembler language. The reader is expected to have a basic knowledge of programming, data processing terms, and fundamental System/3 concepts.

This manual describes the configuration of input/output units, processors, and special features available on System/3 Models 8, 10, 12, and 15. The functions performed by the various models and the lights, switches, and machine level instructions that initiate and control these functions are explained.

The introduction in this manual presents the configurations of features, I/O devices, and main storage capacities for each model. Other chapters and summary charts within this reference manual are organized as follows:

Chapter 1, *Introduction* contains an overview of system characteristics and the physical characteristics of the available I/O devices.

Chapter 2, *Instruction Set* presents the entire instruction set in alphabetic order.

Chapter 3, *Processing Unit* describes the various registers and functions of the processing unit.

Chapter 4, *System Control Panel* discusses the lights, switches and panel procedures for the various models.

The remaining chapters describe the I/O devices available on System/3.

Appendix A contains instruction timings and formats, code conversions, and number charts.

Appendix B defines terms and abbreviations used in this publication.

Related Publications

IBM 1255 Magnetic Character Reader Components Description Manual, GA24-3542

An Introduction to the IBM 3270 Information Display System, GA17-2739

IBM 1403 Printer Component Description, GA24-3073

IBM 1442 Card Read Punch Manual, GA24-3119

IBM 3410/3411 Magnetic Tape Subsystem Component Summary, GA32-0015

IBM 3881 Optical Mark Reader Models 1 and 2 Reference Manual and Operator's Guide, GA21-9143

IBM 3881 Optical Mark Reader Forms Kit, GC20-1750

IBM 3881 Mark Reader Systems Design Guide, GC20-1751

Operator's Guide for IBM 3270 Information Display Systems, GA27-2742

IBM 96-Column Card Reference Manual, GA21-9125

IBM System/3 3741 Reference Manual, GC21-5113

IBM Diskette General Information Manual, GA21-9182

IBM System/3 Model 8 Introduction, GC21-5114

IBM System/3 Card System Introduction, GC21-7505

IBM System/3 Disk System Introduction, GC21-7510

IBM System/3 Model 12 Introduction, GC21-5116

IBM System/3 Model 15 Introduction, GC21-5094

IBM System/3 5448 Disk Storage Drive Program Reference Manual, GC21-5168

Contents

This introduction contains a brief overview of system characteristics you need to be familiar with before you can program System/3. It also presents the available I/O devices, size of storage available, how storage is addressed, the size of the smallest addressable unit, and the types and formats of system instructions.

SYSTEM CONFIGURATIONS BY MODEL

Figures 1-1 through 1-5 show the configurations of input/output devices that are available and list the amount of main storage available for each model. In the figures, solid lines leading from the CPU to the I/O device indicate that the device shown (or one of the devices shown, where two or more devices are shown connected by an OR) is required for the system to operate. Devices attached to the system by dotted lines are available, but are not required.

Program Note: The configurations of required units shown in this manual do not necessarily represent the configurations of I/O devices required on the various System/3 models when IBM programming support is used. This manual assumes that some readers may work with systems not using IBM programming support.

Channel Limitations on Model 10 Configurations

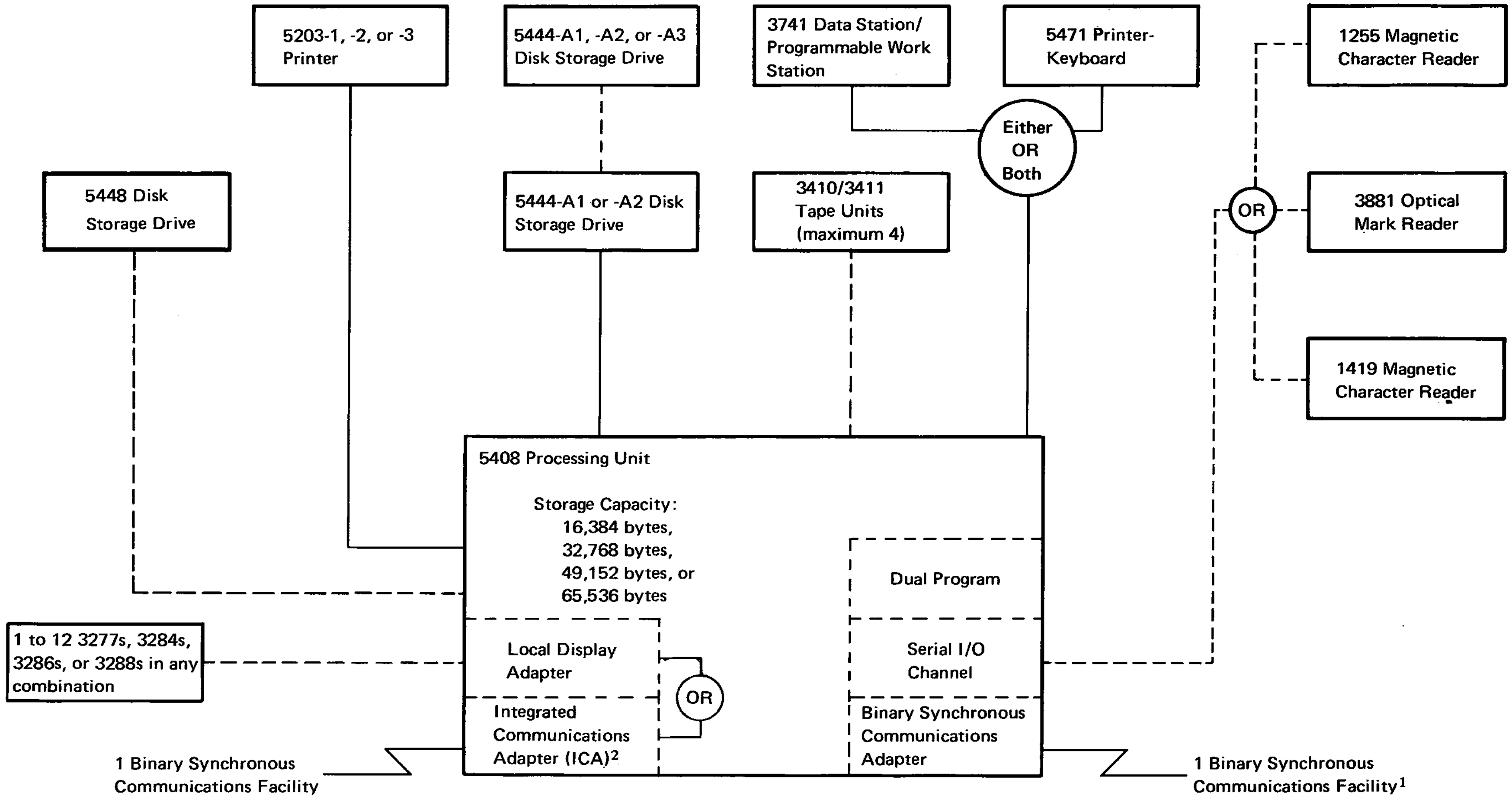
In certain Model 10 system configurations, overlapped I/O operations can cause I/O devices to experience data overrun conditions. Data overrun occurs when requests for I/O cycle steals are not granted in the time limit required for a device. The result of the overrun may cause loss of data and on the 5424 and 5203 this condition is not detected. Therefore, when programming I/O device operations, only those devices should be overlapped which do not cause overrun conditions.

The following chart gives possible configurations:

Devices	Groups of Devices that Avoid Data Overrun – Read Down					
5444	X	X			X	
5445[4]	X		X	X		
5448[4]			X	X		X
5203[1]		X	X			X
1255, 1270, 3881[2]	X	X	X	X	X	X
SIOC (50 KB)		X		X	X	X
3410/3411				X	X	X
BSCA	X	X	X	X	X	X
5424	X	X	X	X	X	X
MLTA	X	X	X	X	X	X
1442	X	X	X	X	X	X
3741	X	X	X	X	X	X
1403[1]	X	X	X	X	X	X
5471, 5475[3]	X	X	X	X	X	X

[1] 5203, 1403 — Mutually exclusive devices.
[2] 1255, 1270, 3881 — SIOC attached devices; they are mutually exclusive and each excludes all other devices on the SIOC.
[3] 5471, 5475 — Mutually exclusive devices.
[4] 5445, 5448 — Mutually exclusive devices.

IBM required special equipment engineering can determine whether configurations involving high data rate devices, such as the RPQ items installed, will result in data overrun if IBM program products are not being used. Contact your IBM sales representative for this information.

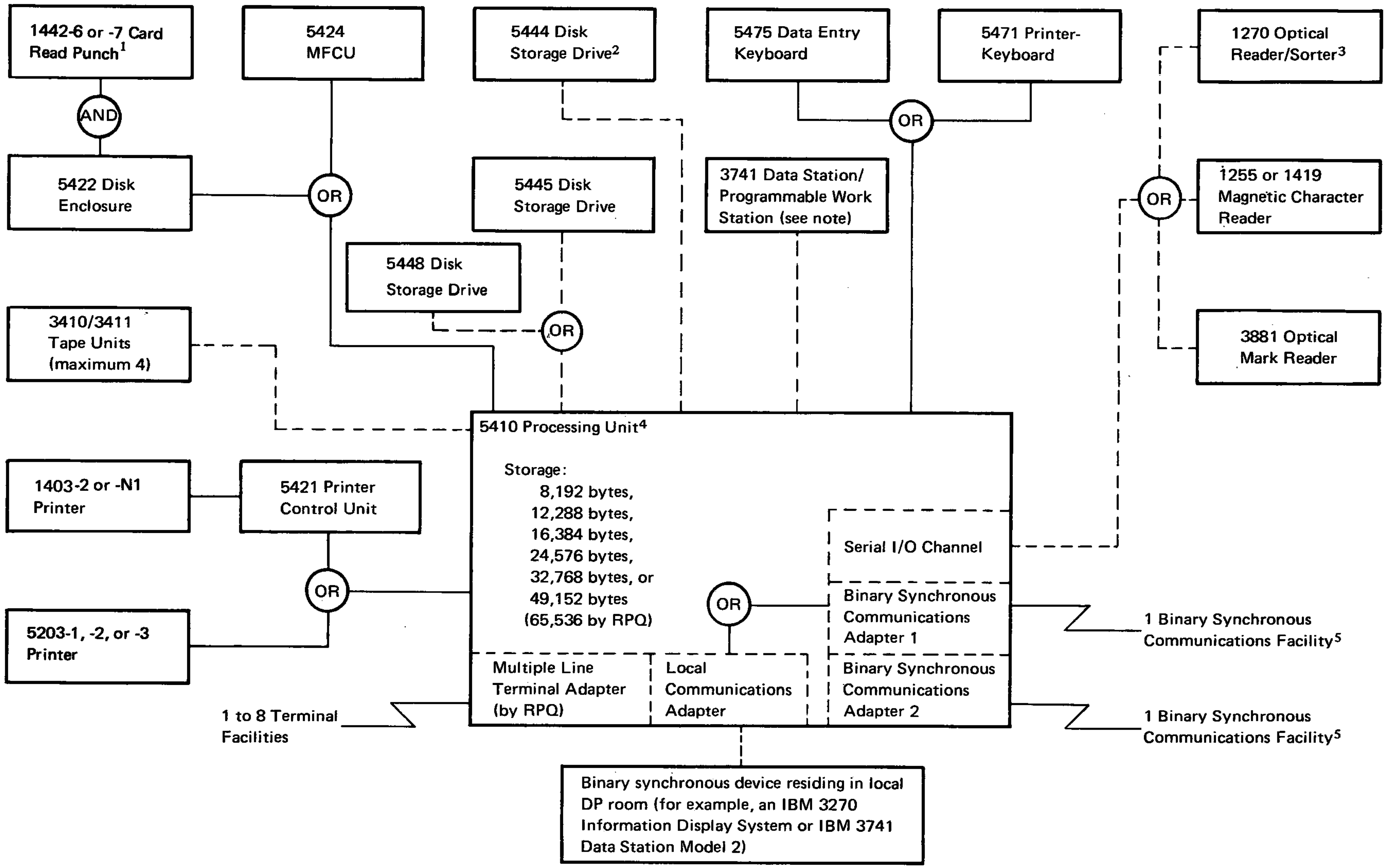


Note: Solid lines indicate required features or devices. Dashed lines indicate features and devices are available, but not required.

[1]The BSCA can be equipped with an EIA Local Attachment Feature that allows a BSC device (such as the IBM 3270 Information Display System) residing in the local DP room to be attached directly to the BSCA without a data set or modem.

[2]3410/3411 and ICA cannot both be installed on the same system.

Figure 1-1. Devices Available for System/3 Model 8



Note: Solid lines indicate required features and devices. Dashed lines indicate features and devices are available, but not required. Only one 3741 can be directly attached to the system.

---

[1] If IBM programming support is used, configurations without the 5424 must include both a 1442 and a 5444, and *no* 5475.

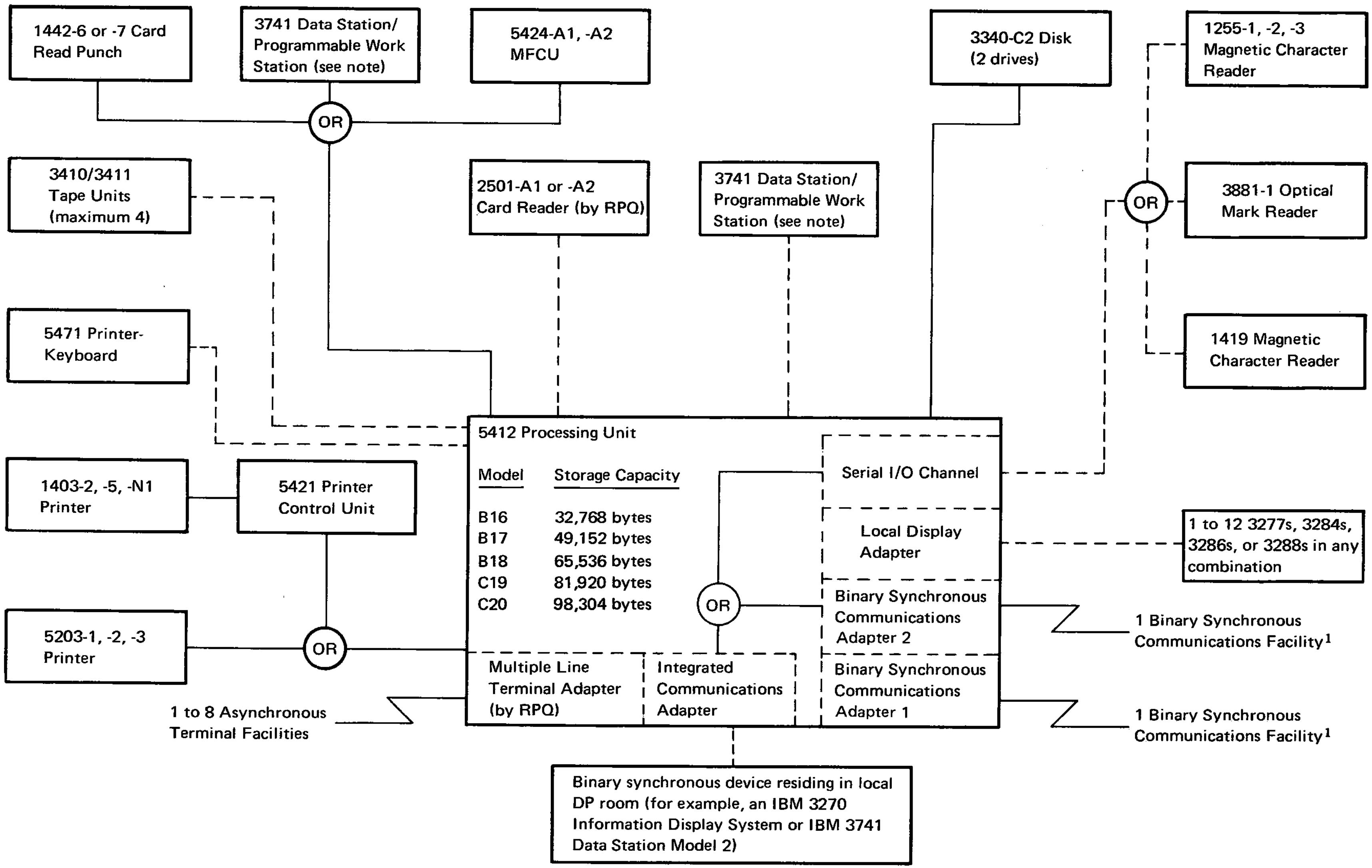
[2] If IBM System/3 disk system programming support is being used, at least one 5444 is required.

[3] Usually not used in the United States.

[4] IBM programming systems for a disk-oriented system require a minimum of 12,288 bytes of storage to ensure systems availability.

[5] The BSCA can be equipped with an EIA Local Attachment Feature that allows a BSC device (such as the IBM 3270 Information Display System) residing in the local DP room to be attached directly to the BSCA without a data set or modem.

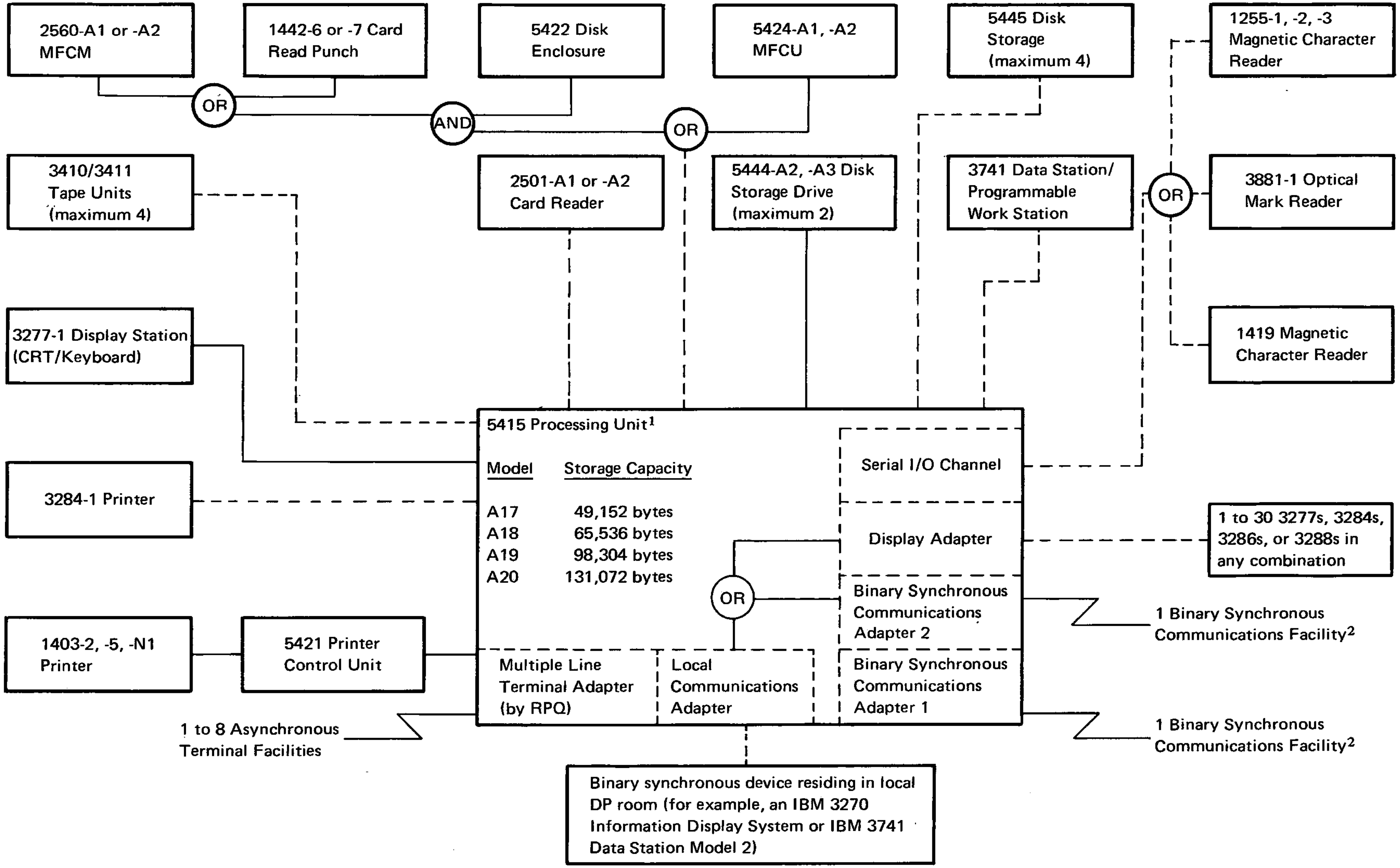
Figure 1-2. Devices Available for System/3 Model 10



Note: Solid lines indicate required features and devices. Dashed lines indicate features and devices are available, but not required. Only one 3741 can be directly attached to the system.

[1]The BSCA can be equipped with an EIA Local Attachment Feature that allows a BSC device (such as the IBM 3270 Information Display System) residing in the local DP room to be attached directly to the BSCA without a data set or modem.

Figure 1-3. Devices Available for System/3 Model 12

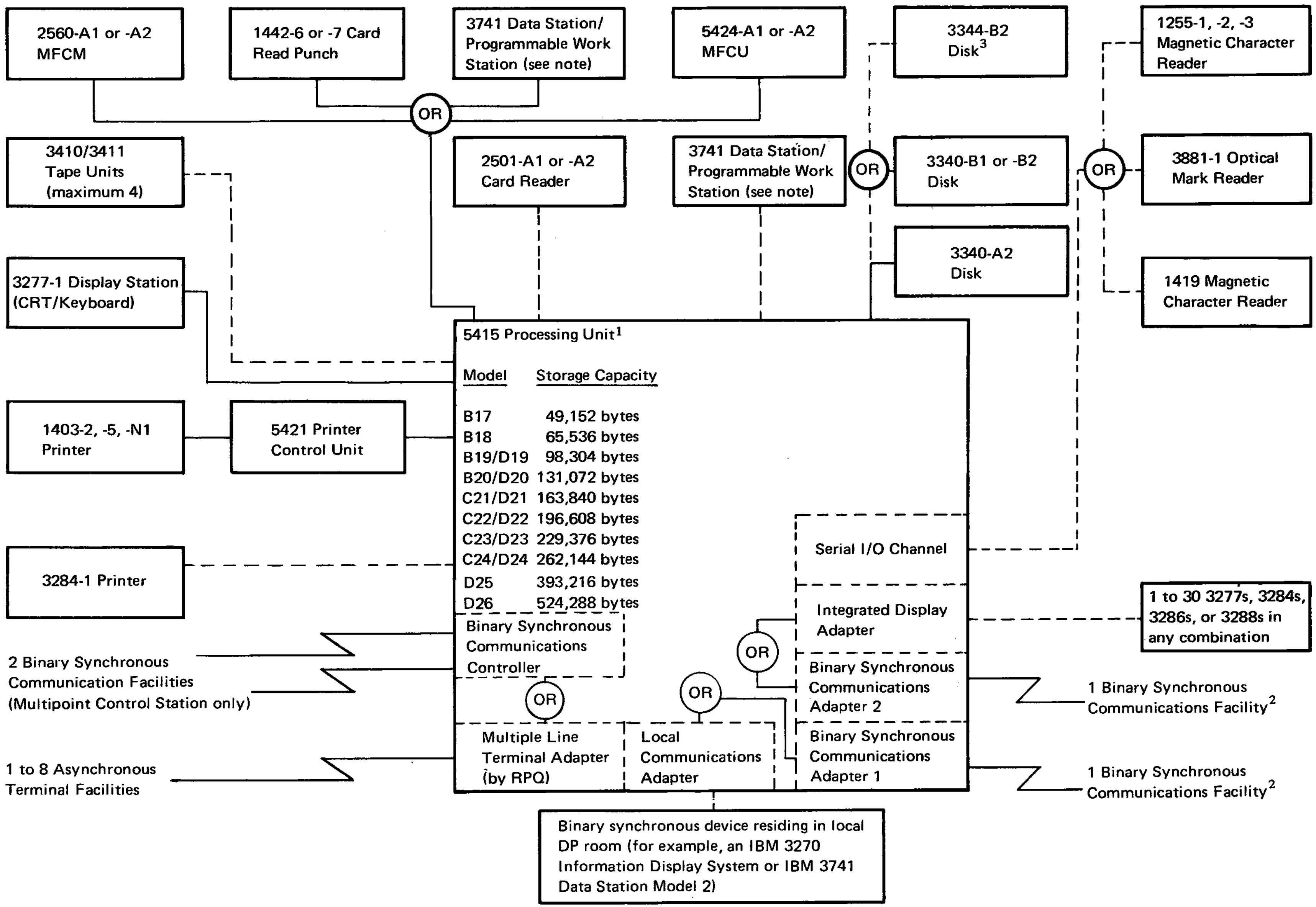


Note: Solid lines indicate required features and devices. Dashed lines indicate features and devices are available, but not required. Only one 3741 can be directly attached to the system.

[1] IBM 5424 required with this unit if IBM programming support is used.

[2] The BSCA can be equipped with an EIA Local Attachment Feature that allows a BSC device (such as the IBM 3270 Information Display System) residing in the local DP room to be attached directly to the BSCA without a data set or modem.

Figure 1-4. Devices Available for System/3 Model 15A



Note: Solid lines indicate required features and devices. Dashed lines indicate features and devices are available, but not required. Only one 3741 can be directly attached to the system.

---

[1] IBM 5424 required with this unit if IBM programming support is used.

[2] The BSCA can be equipped with an EIA Local Attachment Feature that allows a BSC device (such as the IBM 3270 Information Display System) residing in the local DP room to be attached directly to the BSCA without a data set or modem.

[3] The IBM 3344-B2 is available on Models D19 through D24 only.

Figure 1-5. Devices Available for System/3 Models 15B, 15C and 15D

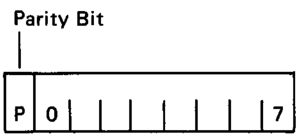
INTERNAL DATA FORMAT

Data Code

Data is stored in System/3 in EBCDIC (extended binary coded decimal interchange code) format.

Byte

The smallest addressable unit in System/3 is the byte, which consists of 8 data bits and a parity bit:



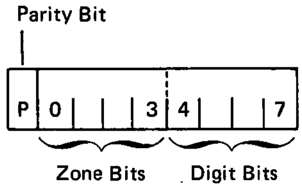
Note that all data in storage looks the same to the processing unit: 8 binary bits. Instructions in the processing unit determine whether the data is treated as zoned decimal, graphic characters, or binary integers.

Parity

System/3 maintains odd parity on data entering the CPU. That is, if a byte entering the CPU has an even number of 1-bits in positions 0 through 7, the CPU places a 1-bit in the parity (P) position to provide an odd number of 1-bits in the byte. Thereafter, anytime a byte is used, the CPU checks to ensure that it contains an odd number of bits (1-bits). If an even number is detected, the CPU stops with a process check.

Zoned Decimal Format

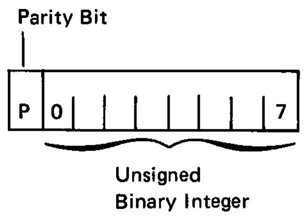
Zoned decimal format divides each byte into two 4-bit groups. Bits 0-3 constitute the zone portion and bits 4-7 constitute the digit portion:



When data is handled in this format, the zone bits do not participate in any arithmetic operations. The zone bits of the low-order byte indicate the sign of the field for arithmetic operations.

Binary Format (Logical Data)

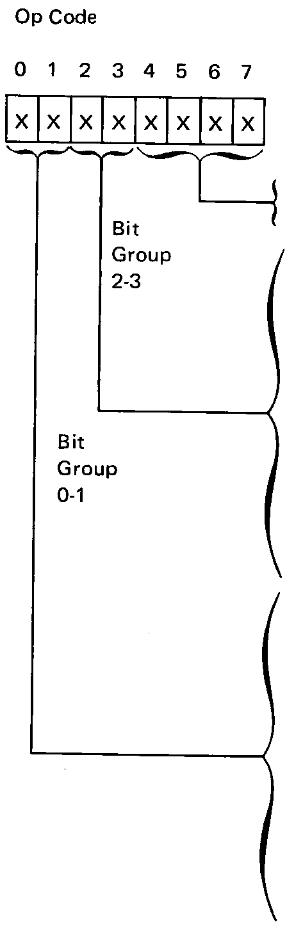
System/3 processes logical data in binary format, treating each byte as an unsigned 8-bit binary integer:



ADDRESSING MAIN STORAGE

Main storage positions are numbered consecutively from hex 0000 to the upper limit of storage. The operand address specifies the rightmost position of the main storage field to be acted upon during instruction execution. The only exception is that operand 1 of the insert-and-test-characters instruction is addressed by its leftmost position.

An instruction can address a main storage location by either of two methods: direct addressing or base-displacement addressing. As shown in Figure 1-6, the first 4 bits in the instruction operation code (op code) specify the type of addressing to be used by the instruction.



Defines type of operation to be performed.

Specifies operand 2 main storage addressing activity for instruction:

- 00 – Operand 2 address portion of the instruction contains a 2-byte address (direct addressing).
- 01 – Content of XR1 (index register 1) is added to the single byte in the operand 2 section of the instruction. The result specifies the storage location being addressed (base-displacement addressing).
- 10 – Content of XR2 (index register 2) is added to the single byte in the operand 2 section of the instruction. The result specifies the storage location being addressed (base-displacement addressing).
- 11 – Operand 2 address portion of the instruction is not used to address main storage in this instruction.

Specifies operand 1 main storage addressing activity for instruction:

- 00 – Operand 1 address portion of the instruction contains a 2-byte address (direct addressing).
- 01 – Content of XR1 is added to the single byte in the operand 1 section of the instruction. The result specifies the storage location being addressed (base-displacement addressing).
- 10 – Content of XR2 is added to the single byte in the operand 1 section of the instruction. The result specifies the storage location being addressed (base-displacement addressing).
- 11 – Operand 1 address portion of the instruction is not used to address main storage in this instruction.

Note: When bits 0, 1, 2, and 3 = 1111 (hex F), the instruction is a command-type instruction and does not address main storage.

Figure 1-6. Op Code Function in Addressing Main Storage

Direct Addressing

When the op code specifies that an operand is being addressed directly by the instruction, the CPU uses the appropriate 2 bytes from the operand address area of the instruction as a direct address of the operand.

Base-Displacement Addressing

When the op code specifies base-displacement addressing for an operand, the CPU adds the immediate data from the single byte in that operand address area to the 2-byte address from the index register specified by the op code, then uses the resulting address as the address of the operand.

The instruction can specify the use of either index register for either operand address. Also, the same index register can be used to determine both operand addresses in the same instruction.

Any one value of an index register allows access to 256 storage positions.

INSTRUCTION FORMATS

System/3 uses three instruction formats of varying length. The type of addressing being performed determines the length of each instruction.

All instruction formats have two elements in common: the op code and the Q-byte. Each of these elements is one byte. The op code determines the type of addressing (thereby the length of the instruction) and the operation to be performed. The function of the Q-byte is determined by the instruction and is discussed with each individual instruction.

Command-Type Instructions

Command-type instructions are always 3 bytes long. In a command-type instruction, the Q-code contains the following information, depending on the instruction:

- Jump condition in jump-on-condition instructions
- Halt identifier (tens position) in halt-program-level instructions
- Feature or function specification in command-CPU instructions
- Device address and function specification in all other command-type instructions

Bits 0-3 of the op code of a command type instruction always contain 1111 (hex F).

Op Code: 1111	Q-Byte	R-Byte

0 3 Bits

One-Address Instructions

One-address instructions can be either 3 or 4 bytes long. These instructions are distinguished by having either bits 0-1 or bits 2-3 of the op code byte both 1's. The 2 bits that are not both 1's can be 01, 10, or 00. If these bits are 00, addressing is direct and the instruction is 4 bytes long. If the bits are 01 or 10, addressing is base displacement; the instruction is 3 bytes long; and index register 1 (01) or index register 2 (10) is used. The Q-byte of a one-address instruction can contain one of the following:

- An immediate operand
- A mask
- A branch condition
- A data selection

One-Address Instruction—Direct Addressing:

Op Code: 0011 1100	Q-Byte	Operand (High-Order Address Byte)	Operand (Low-Order Address Byte)

0 3 Bits

One-Address Instruction—Base-Displacement Addressing:

Op Code: 1110 1101 1011 0111	Q-Byte	Displacement Operand

0 3 Bits

Two-Address Instructions

Two-address instructions can be 4, 5, or 6 bytes long. This instruction type is distinctive in that *neither* bit group 0-1 *nor* bit group 2-3 of the op code byte are both 1's. If all four of bits 0-3 are 0's, addressing is direct, and the instruction is 6 bytes long. If any *one* of bits 0-3 is a 1, one of the addresses is direct, the other address is base displacement, and the instruction is 5 bytes long. If one bit from each of the bit groups is a 1, all addressing is base displacement and the instruction is 4 bytes long.

The index register to be used in base displacement addressing for either operand is determined by the bit in the bit group that is 1. If the bit group = 01, index register 1 is used; if the bit group = 10, index register 2 is used. Both addresses can use the same index register during one instruction.

Two-Address Instruction—Operand 1 Address Direct:

Op Code: 0001 0010	Q-Byte	Operand 1 (High-Order Address Byte)	Operand 1 (Low-Order Address Byte)	Operand 2 Displacement

0 3 Bits

Two-Address Instruction—Operand 2 Address Direct:

Op Code: 0100 1000	Q-Byte	Operand 1 Displacement	Operand 2 (High-Order Address Byte)	Operand 2 (Low-Order Address Byte)

0 3 Bits

Two-Address Instruction—Both Addresses Direct:

Op Code: 0000	Q-Byte	Operand 1 (High-Order Address Byte)	Operand 1 (Low-Order Address Byte)	Operand 2 (High-Order Address Byte)	Operand 2 (Low-Order Address Byte)

0 3 Bits

Two-Address Instruction—Both Addresses Base Displacement:

Op Code: 0101 0110 1001 1010	Q-Byte	Operand 1 Displacement	Operand 2 Displacement

0 3 Bits

96-COLUMN CARD CODE

Systems equipped with the IBM 5424 Multi-Function Card Unit use 96-column cards. Data is stored in these cards in three tiers. Each of these tiers holds 32 columns, with each column containing 6 punch positions. Therefore, the card can contain a maximum of 96 six-bit codes. As the 5424 reads the card, the CPU converts each 6-bit code into 8-bit EBCDIC format. On output to the 5424, the CPU converts the EBCDIC code into 6-bit 96-column card code. For more information about the 96-column card, refer to *IBM 96-Column Card Reference Manual,* GA21-9125.

Eight-Bit Program Card Code (IPL Code)

Six-bit card code limits the number of characters (bit patterns) to 64. The 8-bit bytes used internally in the processing unit allow a maximum of 256 different combinations. The instructions to the system require all of the 256 different combinations. The system provides a method of reading 8 bits into storage while using 6-bit card code.

Eight-bit code uses tier 3 of the card to provide 2 extra bits for each column in tiers 1 and 2. These bits are designated C and D. For tier 1 columns, the 4-bit of the corresponding tier 3 column serves as the C-bit, and the 8-bit serves as the D-bit. For tier 2, the 1-bit of the corresponding tier 3 column serves as the C-bit, and the 2-bit serves as the D-bit. For example, columns 1 and 33 use column 65 for their C- and D-bits, columns 2 and 34 use column 66, etc. Figure 1-7 shows an example of program card code punching. The full card code including the characters that must be punched to obtain 8-bit code are shown in Appendix A.

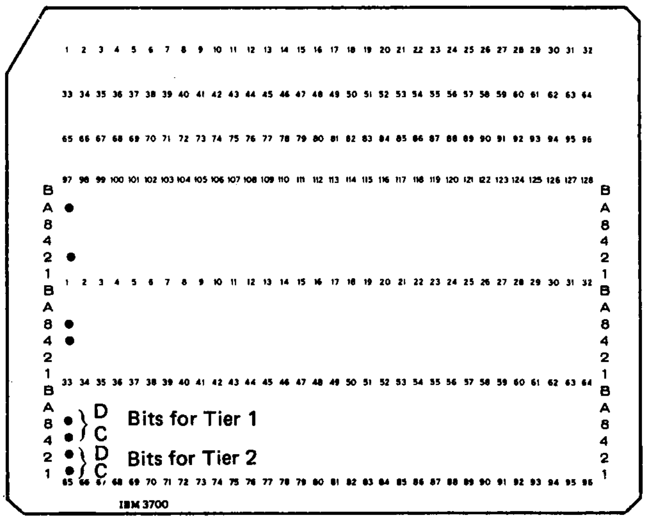


Figure 1-7. Program Card Code Punching

Chapter 2. Instruction Set

This chapter presents the entire System/3 instruction set in alphabetic order. Three of these instructions (command CPU, load CPU, and store CPU) apply only to the Models 12C and 15.

See Appendix A for summary charts that illustrate the various instruction formats and instruction timings.

System/3 uses five types of instructions to handle input and output operations.

- Start I/O
- Sense I/O
- Load I/O
- Test I/O and Branch
- Advance Program Level

The five I/O instructions are presented briefly in this chapter as instructions processed by the CPU. Each chapter in this manual that describes an I/O device includes detailed descriptions of all appropriate I/O instructions for that device.

Although the interval timer and the I/O not-ready-to-ready (unit record restart) features (Model 15 only) are not I/O devices in the usual sense, both are programmed with I/O instructions (Chapter 7). The SIO and TIO instructions used with the dual program feature are also discussed in Chapter 7.

ADD LOGICAL CHARACTERS (ALC)

Op Code (hex)	Q-Byte[1]	Operand Addresses[2]			
Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
0E	L1-1	Operand 1 direct		Operand 2 direct	
1E	L1-1	Operand 1 direct		Op 2 disp from XR1	
2E	L1-1	Operand 1 direct		Op 2 disp from XR2	
4E	L1-1	Op 1 disp from XR1	Operand 2 direct		
5E	L1-1	Op 1 disp from XR1	Op 2 disp from XR1		
6E	L1-1	Op 1 disp from XR1	Op 2 disp from XR2		
8E	L1-1	Op 1 disp from XR2	Operand 2 direct		
9E	L1-1	Op 1 disp from XR2	Op 2 disp from XR1		
AE	L1-1	Op 1 disp from XR2	Op 2 disp from XR2		

[1] L1-1 = number of bytes in operand 1, minus 1
Maximum length of each operand is 256 bytes; both operands must be the same length.

[2] The operands may overlap. Address operands by their rightmost bytes.

Operation

This instruction adds the binary number in operand 2 to the binary number in operand 1 and stores the result in operand 1.

Program Notes

- If operands are overlapped and the operand 1 address is lower than the operand 2 address, data in the overlapped positions of operand 2 is destroyed before it is used in the operation.
- The system resets the binary-overflow bit during this operation.

Condition Register

Bit	Name	Condition Indicated
7	Equal	Zero result
6	Low	No carry occurred from the high-order byte and result not zero
5	High	Carry occurred from the high-order byte and result not zero
4	Decimal overflow	Bit not affected
3	Test false	Bit not affected
2	Binary overflow	Carry occurred from the high-order byte

Example

Instruction:

5E	03	00	10

Index Register 1 = 0CC0

Operand 1 before Operation:

00110101	11001011	11101101	01100100
0CBD	0CBE	0CBF	0CC0

Operand 2:

01011011	01010101	01111000	11001101
0CCD	0CCE	0CCF	0CD0

Operand 1 after Operation:

10010001	00100001	01100110	00110001
0CBD	0CBE	0CBF	0CC0

Condition Register after Operation:

00000010

ADD TO REGISTER (A)

Op Code (hex) Byte 1	Q-Byte[1] Byte 2	Operand Address[2] Byte 3	Byte 4
36	Rx	Operand 1 direct	
76	Rx	Op 1 disp from XR1	
B6	Rx	Op 1 disp from XR2	

[1]Rx specifies the register whose contents are modified by the instruction.
[2]Operand 1 is a 2-byte field addressed by its rightmost byte; the operand is not changed by the operation.

Operation

This instruction adds the unsigned binary number contained in the 2-byte field addressed by the operand address to the contents of the 2-byte register selected by the Q-code. After the addition is complete, the CPU places the sum in the selected register.

The high-order bit (bit 0) of the Q-code specifies which group of registers will be modified. The remaining bits of the Q-code determine which register within the group will be modified.

If bit 0 of the Q-code is 0, bits 1 through 7 specify the register from the following group:

Bit	Register to be Modified
1	Program address recall register on Model 15; program level 2 IAR on Models 8, 10, and 12
2	Program instruction address register on Model 15; program level 1 IAR on Models 8, 10, and 12
3	Instruction address register in use when the add to register instruction is executed
4	Address recall register for current level
5	Program status register
6	Index register 2 (for current level on Models 8, 10, and 12)
7	Index register 1 (for current level on Models 8, 10, and 12)

If the high-order bit of the Q-code is 1, the selected group is the instruction address registers for the interrupt levels. The instruction address registers are selected by the remaining bits as follows:

Bit	Interrupt Level Instruction Address Register	
None	Interrupt level 0	All models
1	Interrupt level 1	All models
2	Interrupt level 2	All models
3	Interrupt level 3	All models
4	Interrupt level 4	All models
5	Interrupt level 5	Model 15 only
6	Interrupt level 6	Model 15 only
7	Interrupt level 7	Model 15 only

This instruction must not be used to add to more than one register at a time. The result of attempting to add to two registers simultaneously can be either incorrect parity or incorrect results in the registers.

This instruction is privileged on Model 15 if bit 0 of the Q-byte equals 1.

Program Note

Even though this instruction can modify the program status register, the contents of the condition register will be placed in the low-order byte of the program status register during I-phase of the next instruction.

The binary overflow bit in the condition register is turned off during I-phase of this instruction.

Condition Register

Bit	Name	Condition Indicated
7	Equal	Zero result
6	Low	No carry occurred from the leftmost byte and result not zero
5	High	Carry occurred from the leftmost byte and result not zero
4	Decimal overflow	Bit not used
3	Test false	Bit not used
2	Binary overflow	Carry occurred from the leftmost byte

Example

Instruction:

36	00000010	00	04

Operand 1:

01001000	00100000
0003	0004

Index Register 2:

Before Operation

00110101	01101010

After Operation

01111101	10001010

Condition Register after Operation:

00000100

ADD ZONED DECIMAL (AZ)

Op Code (hex)	Q-Byte[1]	Operand Addresses[2]			
Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
06	L1-L2 | L2-1	Operand 1 direct		Operand 2 direct	
16	L1-L2 | L2-1	Operand 1 direct		Op 2 disp from XR1	
26	L1-L2 | L2-1	Operand 1 direct		Op 2 disp from XR2	
46	L1-L2 | L2-1	Op 1 disp from XR1	Operand 2 direct		
56	L1-L2 | L2-1	Op 1 disp from XR1	Op 2 disp from XR1		
66	L1-L2 | L2-1	Op 1 disp from XR1	Op 2 disp from XR2		
86	L1-L2 | L2-1	Op 1 disp from XR2	Operand 2 direct		
96	L1-L2 | L2-1	Op 1 disp from XR2	Op 2 disp from XR1		
A6	L1-L2 | L2-1	Op 1 disp from XR2	Op 2 disp from XR2		

[1]L1-L2 (4 bits) = number of bytes in operand 1 minus the number of bytes in operand 2
L2-1 (4 bits) = number of bytes in operand 2, minus 1
Maximum length of operand 1 is 31 bytes; maximum length of operand 2 is 16 bytes.
[2]The operands may overlap. Address operands by their rightmost bytes.

Operation

This instruction algebraically adds the second operand to the first operand and stores the result in the first operand.

The processing unit sets the zone bits of all bytes except the rightmost byte in the first operand to hex F (binary 1111). It sets the zone bits of the rightmost byte in the first operand to (1) hex F if the result of the operation is either positive or zero, or (2) hex D if the result is negative.

Program Notes

- The second operand remains unchanged unless the fields overlap.
- If operands are overlapped and the operand 1 address is lower than the operand 2 address, data in the overlapped positions of operand 2 is destroyed before it is used in the operation.
- The system does not check for valid decimal digits in either operand.
- The decimal overflow condition indicator, which may be set during this operation, is reset by:
  - A system reset
  - Testing decimal overflow with a branch-on-condition or jump-on-condition instruction
  - Loading a 0 in bit 4 of the program status register using the load-register instruction
- The system saves the starting address of operand 1 in the address recall register.

Condition Register

Bit	Name	Condition Indicated
7	Equal	Zero result
6	Low	Negative result
5	High	Positive result
4	Decimal overflow	Carry occurred from the leftmost position of operand 1
3	Test false	Bit not affected
2	Binary overflow	Bit not affected

Example

Instruction:

06	22	00	10	00	20

Operand 1 before Operation:

F7	F6	F3	F6	F9
000C	000D	000E	000F	0010

Operand 2:

F4	F2	F5
001E	001F	0020

Operand 1 after Operation:

F7	F6	F7	F9	F4
000C	000D	000E	000F	0010

Condition Register after Operation:

00000100

ADVANCE PROGRAM LEVEL (APL)

Op Code (hex)	Q-Byte[1]	R-Byte
Byte 1	Byte 2	Byte 3
F1	I	Not used

[1] I (Q-byte) = DA-code, M-code, and N-code, where:
DA (bits 0-3) = the address of the device being tested
M (bit 4) = an address modifier
N (bits 5-7) = a code that specifies the condition for which the device is tested

Operation

This instruction (which is privileged on Model 15) tests the addressed device for the condition specified by the N-code. The operation performed depends on whether the APL is conditional or unconditional.

Conditional APL Instruction: A conditional APL is one that has a DA code other than hex 0. Whenever the DPF (dual program feature) is enabled, the program level advances if the conditions specified by the N-code of the Q-byte exist at the addressed device. The reentry point of the discontinued program level is the starting address of the advance program level instruction. If the specified condition does not exist, no program level advance occurs, and the CPU executes the next sequential instruction.

If the DPF is not installed or is not enabled, a conditional APL instruction causes the program to loop on itself until the tested condition no longer exists at the addressed device. The program then proceeds with the next sequential instruction.

Unconditional APL Instruction: There are two types of unconditional APL instructions: looping and nonlooping.

Nonlooping APL: The nonlooping unconditional APL instruction has an M-code of hex 0 and an N-code of binary 000.

In systems with the DPF enabled, an unconditional program level advance occurs when the CPU executes the instruction. The reentry point to the discontinued level is the next sequential instruction.

If the DPF is not installed or is not enabled, the CPU immediately advances to the next sequential instruction upon encountering a nonlooping unconditional APL.

Looping APL: The looping unconditional APL instruction has an M-code of hex 0 and an N-code of binary 001 through 111.

In systems with the DPF enabled, an unconditional program level advance occurs when the CPU executes the instruction. The reentry point to the discontinued level is the address of the APL instruction.

If the DPF is not installed or is not enabled, the CPU loops on the APL instruction but interrupt routines still function normally. If the unconditional looping APL is executed in an interrupt routine, that routine loops on the APL instruction continually, and only higher priority interrupt routines function normally. To exit from the loop caused by a looping unconditional APL, either modify the IAR of the program or interrupt routine (using either a load-register or add-to-register instruction) or change the instruction residing at the address indicated by the instruction address register (using a move characters instruction, for example).

Program Notes

- APL is a privileged instruction on the Model 15.
- A looping unconditional APL instruction can be used deliberately to prevent the active program from advancing to the next sequential instruction.

Condition Register

This instruction does not affect the condition register.

Example

Instruction:

F1	C8	Not used

Drive 2 Not Ready or Unit Check
5445
APL

BRANCH ON CONDITION (BC)

Op Code (hex)	Q-Byte[1]	Operand Address[2]	
Byte 1	Byte 2	Byte 3	Byte 4
C0	I	Operand 2 direct	
D0	I	Op 2 disp from XR1	
E0	I	Op 2 disp from XR2	

[1]I = binary code specifying condition or conditions that cause branch to occur
[2]Operand portion of instruction specifies the branch-to-address.

Operation

This instruction tests the condition register for the conditions specified by the Q-code. Bit 0 of the Q-code specifies whether the branch is to be performed on condition true (1) or condition false (0). Bit 1 is not used.

Q-Code Bit	Conditions
0 = 0	Branch if all conditions (bits 2 through 7) tested are 0
= 1	Branch if any condition (bits 2 through 7) tested is 1
1	Not used
2	Binary overflow
3	Test false
4	Decimal overflow
5	High
6	Low
7	Equal

Program Notes

- The branch operation performs as a no-op when the Q-code is 80, x7, or xF (where x = 0 through 7).
- An unconditional branch occurs when the Q-byte contains 00, x7, or xF (where x = 8 through F).

Condition Register

Bit	Name	Condition
7	Equal	Bit not affected
6	Low	Bit not affected
5	High	Bit not affected
4	Decimal overflow	Turned off if tested; otherwise not affected
3	Test false	Turned off if tested; otherwise not affected
2	Binary overflow	Not affected

Example

Instruction:

C0	10001000	02	BF
0BCC	0BCD	0BCE	0BCF

Condition Register before Operation:

00011001

Instruction Address Register after Operation:

02	BF

Address Recall Register after Operation:

0B	D0

Condition Register after Operation:

00010001

COMMAND CPU (CCP) – MODELS 12C AND 15

Op Code (hex) Byte 1	Q-Byte[1] Byte 2	R-Byte[2] Byte 3
F4	Rx	I1

[1]Rx = ID codes for the function to be acted upon
[2]I1 = code specifying action to occur

Operation

The processing unit performs the actions specified by the R-byte with the functions specified by the Q-byte (Figure 2-1). Command CPU is a privileged instruction (on Model 15 only).

The Model 12C assembler does not recognize the mnemonic CCP but the hardware recognizes the op code for this command.

To use this command for a Model 12C assembler, the programmer has the following options:

- Code the instruction using DCs
- Use $CCP macro (see *IBM System/3 Models 8, 10, and 12 System Control Programming Macros Reference Manual*, GC21-7562)

Program Notes, Diagnostic Mode

- Mode change is effective at the next sequential instruction (NSI).
- System reset/IPL resets diagnostic mode.
- This instruction performs as a no op on Models 12C, 15A, 15B, and 15C.
- This instruction cannot modify the I/O >256 bit of the PMR (LCP must be used).

Condition Register

This instruction does not affect the condition register.

Example

Instruction:

F4	30	08

Program Mode Register (Current) before Operation:

74

Program Mode Register (Current) after Operation:

08

Q-Byte	Function	Control Code	Action
10[5]	Supervisor call (SVC)[1]	00	Request interrupt level 0
		02	Reset interrupt level 0
20[5]	Program check interrupt control[2]	00	Disable interrupt level 7
		01	Enable interrupt level 7
		02	Reset and disable interrupt level 7
		03	Reset and enable interrupt level 7
30	Load current program mode register (PMR) immediate[3]	Bit 0	I/O greater than 128K
		Bit 1	EB cycle address translate
		Bit 2	EA cycle address translate
		Bit 3	I cycle address translate
		Bit 4	Privileged state
		Bit 5	I/O greater than 64K
		Bit 6	Protection state
		Bit 7	Mask interrupt state
40[5]	Model 15D diagnostic mode[4]	00	Set diagnostic mode (slow)
		02	Reset diagnostic mode (fast)

[1]The supervisor call command allows the program to force the CPU to enter interrupt level 0. If bit 6 of the control code is 0, if no higher interrupt priority exists, and if the mask interrupt bit in the current PMR is not on, the CPU enters interrupt level 0 at the completion of the supervisor-call instruction. While in 0, the system is in privileged mode and all commands can be executed. (Therefore, while in level 0 the program can change the system status by altering the PMRs.) Essentially, the supervisor call is the means of communication between the unprivileged application programs and the privileged supervisor program residing at interrupt level 0. Level 0 is always enabled and cannot be disabled. A command CPU instruction with a Q-code of hex 10 is not a privileged instruction.

[2]Interrupt level 7 must be enabled to use the program check interrupt feature. Once enabled, an invalid address, invalid operation, storage violation, privileged operation in non-privileged mode, or invalid Q-byte will cause a program check interrupt. (If level 7 is not enabled, these conditions will cause a processor check and the system will stop.) The interrupt 7 routine must store the program check address register and the program check status register before level 7 is reset, because resetting level 7 resets these registers also.

[3]Load current PMR immediate provides another way to change the current PMR (the PMR that is selected by the interrupt level when the command CPU instruction is issued). The control code is treated as immediate data and is loaded into the current PMR. These new contents become effective at NSI (next sequential instruction) if the active program level is not changed via an interrupt.

[4]Model 15D system diagnostics require certain attachment and device diagnostics to be executed at normal System/3 rate. This instruction allows fast/slow rate of CPU operation.

[5]Not available on Model 12C.

Figure 2-1. Command CPU Functions and Actions

COMPARE LOGICAL CHARACTERS (CLC)

Op Code (hex)	Q-Byte[1]	Operand Addresses[2]			
Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
0D	L1-1	Operand 1 direct		Operand 2 direct	
1D	L1-1	Operand 1 direct		Op 2 disp from XR1	
2D	L1-1	Operand 1 direct		Op 2 disp from XR2	
4D	L1-1	Op 1 disp from XR1	Operand 2 direct		
5D	L1-1	Op 1 disp from XR1	Op 2 disp from XR1		
6D	L1-1	Op 1 disp from XR1	Op 2 disp from XR2		
8D	L1-1	Op 1 disp from XR2	Operand 2 direct		
9D	L1-1	Op 1 disp from XR2	Op 2 disp from XR1		
AD	L1-1	Op 1 disp from XR2	Op 2 disp from XR2		

[1] L1-1 = number of bytes in operand 1 minus 1
Maximum length of each operand is 256 bytes; both operands must be the same length.

[2] The operands may overlap. Address operands by their rightmost bytes.

Operation

This instruction compares operand 1 to operand 2, byte by byte, and sets the condition register according to the result of the comparison. The comparison treats each operand as a binary quantity; that is, corresponding bytes from the two operands are compared, bit for bit.

Program Note

Neither operand is altered by the instruction.

Condition Register

Bit	Name	Condition Indicated
7	Equal	Operand values are equal
6	Low	Operand 1 value smaller than operand 2 value
5	High	Operand 1 value greater than operand 2 value
4	Decimal overflow	Bit not affected
3	Test false	Bit not affected
2	Binary overflow	Bit not affected

Example

Instruction:

0D	02	00	12	00	02

Operand 1:

27	FA	26
0010	0011	0012

Operand 2:

23	FA	26
0000	0001	0002

Condition Register:

00000100

COMPARE LOGICAL IMMEDIATE (CLI)

Op Code (hex) Byte 1	Q-Byte[1] Byte 2	Operand Address[2] Byte 3	Byte 4
3D	I	Operand 1 direct	
7D	I	Op 1 disp from XR1	
BD	I	Op 1 disp from XR2	

[1] I = immediate binary data to be compared with addressed operand data
[2] Addressed operand is single byte of storage to be compared with Q-byte data.

Operation

This instruction compares the binary immediate operand contained in the Q-byte to the binary operand in storage located at the operand address; the result sets the condition register. Neither operand is changed as a result of this operation.

Condition Register

Bit	Name	Condition Indicated
7	Equal	Operand 1 value equal to Q-byte value
6	Low	Operand 1 value less than Q-byte value
5	High	Operand 1 value greater than Q-byte value
4	Decimal overflow	Bit not affected
3	Test false	Bit not affected
2	Binary overflow	Bit not affected

Example

Instruction:

3D	7F	00	21

Storage Operand:

7F

0021

Condition Register after Operation:

00000001

EDIT (ED)

Op Code (hex)	Q-Byte[1]	Operand Addresses[2]			
Byte 1	**Byte 2**	**Byte 3**	**Byte 4**	**Byte 5**	**Byte 6**
0A	L1-1	Operand 1 direct		Operand 2 direct	
1A	L1-1	Operand 1 direct		Op 2 disp from XR1	
2A	L1-1	Operand 1 direct		Op 2 disp from XR2	
4A	L1-1	Op 1 disp from XR1	Operand 2 direct		
5A	L1-1	Op 1 disp from XR1	Op 2 disp from XR1		
6A	L1-1	Op 1 disp from XR1	Op 2 disp from XR2		
8A	L1-1	Op 1 disp from XR2	Operand 2 direct		
9A	L1-1	Op 1 disp from XR2	Op 2 disp from XR1		
AA	L1-1	Op 1 disp from XR2	Op 2 disp from XR2		

[1] L1-1 = length operand 1 minus 1, in bytes
[2] Operand 2 contains the same number of bytes as operand 1 contains hex 20's.

Operation

The decimal numeric characters in the second operand replace the bytes containing hex 20 in the edit pattern contained in the first operand. All characters other than hex 20 in the edit pattern remain unchanged. The zone bits of all the replaced characters are set to all 1's. The result of the edit operation occupies the first operand. The second operand is not changed. Address the operands by their rightmost bytes. The operands cannot be overlapped.

Condition Register

Bit	Name	Condition Indicated
7	Equal	Operand 2 zero
6	Low	Operand 2 negative
5	High	Operand 2 positive
4	Decimal overflow	Bit not affected
3	Test false	Bit not affected
2	Binary overflow	Bit not affected

Example

Instruction:

0A	0A	00	BF	00	07

Operand 1 before Operation:

$	20		20	20	20
00B5	00B6	00B7	00B8	00B9	00BA

.	20	20	ƀ	*
00BB	00BC	00BD	00BE	00BF

Operand 2:

0	1	0	8	0	R
0002	0003	0004	0005	0006	0007

Note: R represents -9

Operand 1 after Operation:

$	0		1	0	8
00B5	00B6	00B7	00B8	00B9	00BA

	0	9	ƀ	*
00BB	00BC	00BD	00BE	00BF

Note: Location 00BD contains a 9 because the zone bits of all replaced characters (0's) in the edit pattern are set to all 1's.

Condition Code:

00000010

HALT PROGRAM LEVEL (HPL)

Op Code (hex)	Q-Byte[1]	R-Byte[2]
Byte 1	Byte 2	Byte 3
F0	I2	I1

[1]I2 = hex code of the left character in the 2-character halt identifier
[2]I1 = hex code of the right character in the halt identifier

Operation

This instruction prevents execution of the next sequential instruction by:

1. Branching on itself if the system is not using the dual program feature, or

2. Causing an advance to the alternate program level when the system is equipped with the dual program feature and that feature is enabled.

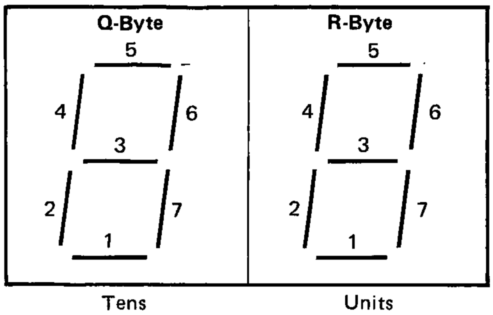
When a halt or branch occurs, the CPU displays a halt identifier code on a display unit on the system control panel.

Pressing the system START/HALT RESET key while the program level is in the halt state resets the halt state and its associated halt identifier code, restarting the interrupted level at its next sequential instruction. If a second programmed halt occurs (on the alternate level) before the first programmed halt has been serviced, the CPU loops on the second program level instruction until the first programmed halt has been serviced, then executes the second halt.

The message display unit consists of 14 bar lights arranged as shown in Figure 2-2.

The hex digits required in a byte to produce the common characters used as halt identifiers are shown in Figure 2-3.

Figure 2-4 shows how a 2 is formatted on the display and how the code that specifies a 2 is generated by combining bits that turn on the required bar lights.



Bits 1-7 turn on bar lights 1-7, respectively.

Figure 2-2. Message Indicator Light Arrangement

CHARACTER	HEX CODE	DISPLAY SEEN	CHARACTER	HEX CODE	DISPLAY SEEN
None	00		A	3F	
1	03		b	79	
2	76		C	6C	
3	57		d	73	
4	1B		E	7C	
5	5D		F	3C	
6	7D		H	3B	
7	07		J	63	
8	7F		L	68	
9	5F		P	3E	
0	6F		U	6B	

Figure 2-3. Coding for Typical Halt Identifier Characters

Bits – 0 1 2 3 4 5 6 7
= State – X 1 1 1 0 1 1 0

Hex Codes 7 6

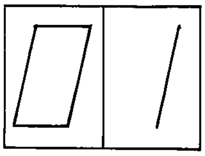
Figure 2-4. Relationship between Bars Used and Hex Code

Example

Instruction:

F0	6F	03

Display Unit:



Program Notes

- The halt program level instruction is not executed when it is used in an interrupt level program sequence.
- The program level can be stopped with a halt program level instruction to wait for an interrupt request. The interrupt routine can modify an appropriate program level instruction address register with a load register instruction to return to the halted program level at an instruction other than the halt instruction. The halted program level resumes operation after the interrupt is reset. The display unit is turned off by a load register instruction. The program level resumes operation according to normal priority.
- The halt instruction is a privileged instruction (on Model 15 only).

Condition Register

This instruction does not affect the condition register.

INSERT AND TEST CHARACTERS (ITC)

Op Code (hex) Byte 1	Q-Byte[1] Byte 2	Operand Addresses[2] Byte 3	Byte 4	Byte 5	Byte 6
0B	L1-1	Operand 1 direct		Operand 2 direct	
1B	L1-1	Operand 1 direct		Op 2 disp from XR1	
2B	L1-1	Operand 1 direct		Op 2 disp from XR2	
4B	L1-1	Op 1 disp from XR1	Operand 2 direct		
5B	L1-1	Op 1 disp from XR1	Op 2 disp from XR1		
6B	L1-1	Op 1 disp from XR1	Op 2 disp from XR2		
8B	L1-1	Op 1 disp from XR2	Operand 2 direct		
9B	L1-1	Op 1 disp from XR2	Op 2 disp from XR1		
AB	L1-1	Op 1 disp from XR2	Op 2 disp from XR2		

[1] L1-1 = number of bytes in operand 1, minus 1
[2] Address operand 1 by its leftmost position. Operand 2 is a single-byte field.

Operation

The single character at the operand 2 address replaces all the characters to the left of the first significant digit in operand 1. Only the decimal digits 1 through 9 are significant.

For example, if the leftmost byte of a field to be printed contains a dollar sign, the first operand address should be the address of the byte to the right of the dollar sign.

The operation proceeds from left to right. Filling operand 1 with the character from operand 2 or encountering a significant digit in operand 1 ends the operation.

Program Notes

- Operand 2 remains unchanged.
- At the end of this operation, the address recall register contains the address of the first significant digit of operand 1; if no significant digit is found, it contains the address of the byte to the right of operand 1. This new information remains in the register until the system executes the next decimal-add, decimal-subtract, branch, test-I/O-and-branch, or insert-and-test-character instruction.

Condition Register

This instruction does not affect the condition register.

Example

Instruction:

0B	09	00	B6	00	10

Operand 1 before Operation:

$	0	,	1	0	8
00B5	00B6	00B7	00B8	00B9	00BA

.	0	9	ƀ	*
00BB	00BC	00BD	00BE	00BF

Operand 2:

*
0010

Operand 1 after Operation:

$	*	*	1	0	8
00B5	00B6	00B7	00B8	00B9	00BA

.	0	9	ƀ	*
00BB	00BC	00BD	00BE	00BF

Note that address 00B5 was not included in the first operand.

JUMP ON CONDITION (JC)

Op Code (hex)	Q-Byte[1]	R-Byte[2]
Byte 1	Byte 2	Byte 3
F2	I	Control code

[1] I = 8 bits of information specifying condition register bits to be tested and the conditions under which a jump is to occur
[2] R-byte contains number (0-255) of bytes that is added to the address of the next sequential instruction (the value in the IAR) to specify the jump to address.

Operation

This instruction tests the condition register for the condition or conditions specified by the Q-code. If the condition register satisfies the condition or conditions established by the Q-byte, the 1-byte control code is added to the value in the instruction address register (the address of the next sequential instruction), and the sum becomes the address of the next instruction.

When bit 0 of the Q-byte = 1, the jump occurs on condition true; when bit 0 = 0, the jump occurs on condition false.

Bits 2 through 7 of the Q-byte define the condition register bits to be tested. More than one condition register bit can be tested at the same time. The Q-byte bits and the conditions tested are:

Q-Code Bit	Conditions
0 = 0	Jump if all conditions tested are 0 (bits 2-7)
= 1	Jump if any condition tested is 1 (bits 2-7)
1	Not used
2	Binary overflow
3	Test false
4	Decimal overflow
5	High
6	Low
7	Equal

Program Notes

- The jump operation performs as a no-op when the Q-code 80, x7, or xF (where x = 0 through 7)
- An unconditional jump occurs when the Q-code is 00, x7, or xF (where X = 8 through F).

Condition

Bit	Name	Condition Indicated
7	Equal	Bit not affected
6	Low	Bit not affected
5	High	Bit not affected
4	Decimal overflow	Turned off if tested; otherwise not affected
3	Test false	Turned off if tested; otherwise not affected
2	Binary overflow	Bit not affected

Example

Instruction:

F2	00110000	0F
0BBD	0BBE	0BBF

Condition Register before Operation:

00001001

Instruction Address Register after Operation:

0B	CF

Condition Register after Operation:

00001001

LOAD ADDRESS (LA)

Op Code (hex)	Q-Byte[1]	Operand Address[2]	
Byte 1	Byte 2	Byte 3	Byte 4
C2	Rx	Operand 2 direct	
D2	Rx	Op 2 disp from XR1	
E2	Rx	Op 2 disp from XR2	

[1]Rx specifies the index register to be loaded:
XR1 = hex 01
XR2 = hex 02

[2]A direct address is loaded when the instruction has a C2 op code. When the op code is D2, the system adds the instruction byte 3 value to the contents of XR1 and stores the result in the index register specified by the Q-byte. When the op code is E2, the system adds the instruction byte 3 value to the contents of XR2 and stores the result in the index register specified by the Q-byte.

Operation

This instruction loads the value specified by instruction byte 3 or instruction bytes 3 and 4 into the index register specified by the Q-byte.

Condition Register

This instruction does not affect the condition register.

Example

Instruction:

D2	02	05

Index Register 1:

BA	15

Index Register 2 after Operation:

BA	1A

LOAD CPU (LCP) – MODELS 12C AND 15

Op Code (hex) Byte 1	Q-Byte[1] Byte 2	Operand Address[2] Byte 3	Byte 4
3F	Rx	Operand 1 direct	
7F	Rx	Op 1 disp from XR1	
BF	Rx	Op 1 disp from XR2	

[1] Rx = register into which data is to be loaded, expressed in binary
[2] Operand addressed is a 2-byte field.

Operation

The CPU moves the data from the 2-byte field specified by the operand address to the register specified by the Q-byte of the instruction. The Q-codes that can be used by the programmer are the same as those specified for the store CPU instruction. Three other Q-codes are accepted by the system for CE diagnostic functions: hex 21, hex 22 and hex 23.

The Model 12C assembler does not recognize the mnemonic LCP but the hardware recognizes the op code for this command.

To use this command for a Model 12C assembler, the programmer has the following options:

- Code the instruction using DCs
- Use $LCP macro (see *IBM System/3 Models 8, 10, and 12 System Control Programming Macros Reference Manual*, GC21-7562)

Program Notes

- The program notes for the store CPU command apply to the load CPU command.
- On the Model 15, load CPU instructions that use Q-codes 20, 21, 22, 23, and 30 are diagnostic commands and are not functional if the program check interrupt (level 7) is enabled.
- On the Model 12C, load CPU instructions that use Q-codes 12 through 17; 1D through 3F, and 41 through FF are invalid.
- On the Model 12C, load CPU instructions that use Q-code 10, address PMR Program Level 1. Load CPU instructions that use Q-code 11, address PMR Program Level 2.
- The low-order byte (operand address minus 1) is used to set the I/O >256K bit (byte 2, bit 7) on Model 15 D25 and Model 15 D26.

Condition Register

This instruction does not affect the condition register.

Example

Instruction:

3F	10	00	39

Operand:

0038	0039
01	4A

Program Level Program Mode Register before Operation:

00	7F

Model 15 D25 and Model 15 D26 only

Program Level Program Mode Register after Operation:

01	4A

Model 15 D25 and Model 15 D26 only

LOAD I/O (LIO)

Op Code (hex)	Q-Byte[1]	Operand Address[2]	
Byte 1	Byte 2	Byte 3	Byte 4
31	I	Operand 1 direct	
71	I	Op 1 disp from XR1	
B1	I	Op 1 disp from XR2	

[1] I (Q-byte) = DA-code, M-code, and N-code, where:
DA (in bits 0-3) = address of device with register being loaded
M (in bit 4) = an address modifier
N (in bits 5-7) = a code that specifies the register being loaded
[2] The operand holds 2 bytes of data, and is addressed by its rightmost (higher numbered) position.

Operation

This operation loads 2 bytes of data from the main storage location specified by the operand address into the destination specified by the N-code. At the same time, when an I/O LSR is specified as the destination on a Model 15, the CPU also loads the I/O > 64K bit into the LSR. When the Model 15 is equipped with more than 128K or more than 256K of storage, the CPU loads the I/O > 128K bit or the I/O > 256K bit into the LSR. These bits must be loaded into the PMR before issuing the LIO instruction.

Program Notes

- If the system is not equipped with the dual program feature or if that feature is not enabled, a load I/O instruction to a busy or not-ready device causes the program to loop at the load I/O instruction until the device becomes not busy or ready. If the system is equipped with a dual program feature and if that feature is enabled, a load I/O instruction to a busy or not-ready device causes a program level advance. When the interrupted level becomes active again, its program resumes at the beginning of the LIO instruction.
- Load I/O is a privileged instruction on the Model 15.
- A Q-byte of hex 00 results in a no-op condition. The CPU accesses the next sequential instruction without performing the load function.

Condition Register

This instruction does not affect the condition register.

Example

Instruction:

31	CC	2F	B1

2F B1 — Location of Data Address
C — Disk Drive 2 DDAR
C — 5445
31 — Load I/O

Operand:

0F	00
2FB0	2FB1

Data Address

Disk Data Address Register before Operation:

0B	20

No Meaning

Disk Data Address Register after Operation:

0F	00

Address of Data Field

LOAD REGISTER (L)

Op Code (hex)	Q-Byte[1]	Operand Address[2]	
Byte 1	Byte 2	Byte 3	Byte 4
35	Rx	Operand 1 direct	
75	Rx	Op 1 disp from XR1	
B5	Rx	Op 1 disp from XR2	

[1]Rx specifies the register into which data is to be loaded.
[2]Operand 1 is a 2-byte field addressed by its rightmost byte; operand 2 is not used.

Operation

This instruction places the contents of the 2-byte field specified by the operand address into the local storage register specified by the Q-byte. The Q-byte specifications discussed for the store-register instruction apply to this instruction. The operation does not alter the operand.

Program Notes

- Do not load more than one register at a time.

- When the program status register (Model 15 only) is selected, the contents of the rightmost byte of the operand have the following significance:

  Bit 7 = 1: Set equal condition
  Bit 6 = 1: Set low condition if bit 7 = 0
  Bit 6 = 0: Set high condition if bit 7 = 0
  Bit 4 = 1: Set decimal overflow condition
  Bit 3 = 1: Set test false condition
  Bit 2 = 1: Set binary overflow condition

  The processing unit ignores the leftmost byte of the operand and bits 0, 1, and 5 of the rightmost byte of the operand.

  The condition register is set at the same time as the program status register under these same controls.

- If the program level has been halted and this instruction is used by an interrupt routine to load the program level instruction address register, the program level will be reset from the halt state and will proceed after all interrupts and I/O cycle steals have been serviced. The program level halt indicators will be turned off.

Condition Register

This instruction does not affect the condition register setting, unless the program status register is specified.

Example

Instruction:

35	00000100	00	11

Operand:

00000000	00000000
0010	0011

Program Status Register before Operation:

00001100	00110001

Program Status Register after Operation:

00000000	00000100

Condition Register after Operation:

00000100

MOVE CHARACTERS (MVC)

Op Code (hex) Byte 1	Q-Byte[1] Byte 2	Operand Addresses[2] Byte 3	Byte 4	Byte 5	Byte 6
0C	L1-1	Operand 1 direct		Operand 2 direct	
1C	L1-1	Operand 1 direct		Op 2 disp from XR1	
2C	L1-1	Operand 1 direct		Op 2 disp from XR2	
4C	L1-1	Op 1 disp from XR1	Operand 2 direct		
5C	L1-1	Op 1 disp from XR1	Op 2 disp from XR1		
6C	L1-1	Op 1 disp from XR1	Op 2 disp from XR2		
8C	L1-1	Op 1 disp from XR2	Operand 2 direct		
9C	L1-1	Op 1 disp from XR2	Op 2 disp from XR1		
AC	L1-1	Op 1 disp from XR2	Op 2 disp from XR2		

[1] L1-1 = number of bytes in operand 1 minus 1
[2] Maximum length of each operand is 256 bytes; both operands must be the same length. The operands may overlap. Address operands by their rightmost byte.

Operation

This instruction places the contents of operand 2, byte by byte, into operand 1. It is possible to propagate one character through an entire field by setting the operand 2 address 1 byte to the right of the operand 1 address.

Program Notes

- The second operand remains unchanged unless the fields overlap.
- If operands are overlapped and the operand 1 address is lower than the operand 2 address, data in the overlapped positions of operand 2 is destroyed before it is used in the operation.

Condition Register

This instruction does not affect the condition register.

Example

Instruction:

0C	05	1A	06	2B	5A

Operand 1 before Operation:

D1	C1	D4	C5	E2	40
1A01	1A02	1A03	1A04	1A05	1A06

Operand 2:

D9	D6	C2	C5	D9	E3
2B55	2B56	2B57	2B58	2B59	2B5A

Operand 1 after Operation:

D9	D6	C2	C5	D9	E3
1A01	1A02	1A03	1A04	1A05	1A06

MOVE HEX CHARACTER (MVX)

Op Code (hex)	Q-Byte[1]	Operand Addresses[2]			
Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
08	I	Operand 1 direct		Operand 2 direct	
18	I	Operand 1 direct		Op 2 disp from XR1	
28	I	Operand 1 direct		Op 2 disp from XR2	
48	I	Op 1 disp from XR1	Operand 2 direct		
58	I	Op 1 disp from XR1	Op 2 disp from XR1		
68	I	Op 1 disp from XR1	Op 2 disp from XR2		
88	I	Op 1 disp from XR2	Operand 2 direct		
98	I	Op 1 disp from XR2	Op 2 disp from XR1		
A8	I	Op 1 disp from XR2	Op 2 disp from XR2		

[1] I = one byte of immediate data that specifies which portion of each single-byte operand is used in the operation
[2] Both operands are single-byte fields.

Operation

This instruction moves the numeric portion (bits 4-7) or the zone portion (bits 0-3) of the second operand to the numeric or zone portion of the first operand, as specified by the Q-byte. Q-byte coding is:

Hex	Binary	Meaning
00	0000 0000	Move data from operand 2 zone portion to operand 1 zone portion
01	0000 0001	Move data from operand 2 numeric portion to operand 1 zone portion
02	0000 0010	Move data from operand 2 zone portion to operand 1 numeric portion
03	0000 0011	Move data from operand 2 numeric portion to operand 1 numeric portion

Program Notes

- The six leftmost binary bits in the Q-byte should be 0's.
- The second operand is not changed unless the same byte is used for both operands.

Condition Register

The condition register is not affected by this instruction.

Example

Instruction:

98	01	A0	65

Index Register 1 = 2B15
Index Register 2 = 1F20

Operand 1 before Operation:

2F

1FC0

Operand 2:

4C

2B7A

Operand 1 after Operation:

CF

1FC0

MOVE LOGICAL IMMEDIATE (MVI)

Op Code (hex) Byte 1	Q-Byte[1] Byte 2	Operand Address[2] Byte 3	Byte 4
3C	I	Operand 1 direct	
7C	I	Op 1 disp from XR1	
BC	I	Op 1 disp from XR2	

[1] I = one byte of immediate data (for example, one byte of actual data or a single-byte mask)
[2] Operand 1 is a single-byte field; operand 2 is not used.

Operation

This instruction moves the Q-byte into operand 1.

Condition Register

This instruction does not affect the condition register.

Example

Instruction:

3C	AF	2F	CB

Operand before Operation:

00

2FCB

Operand after Operation:

AF

2FCB

SET BITS OFF MASKED (SBF)

Op Code (hex)	Q-Byte[1]	Operand Address[2]	
Byte 1	Byte 2	Byte 3	Byte 4
3B	I	Operand 1 direct	
7B	I	Op 1 disp from XR1	
BB	I	Op 1 disp from XR2	

[1]The Q-byte contains a single-byte binary mask specifying operand bits to be turned off.
[2]Operand 1 is a single-byte field; operand 2 is not used.

Operation

The system examines the Q-byte, bit by bit. Whenever it encounters a binary 1 in the Q-byte, the system sets the corresponding bit in the operand byte to 0; whenever it encounters a binary 0 in the Q-byte, it leaves the corresponding bit in the operand unchanged.

Condition Register

This instruction does not affect the condition register.

Example

Instruction:

3B	10000001	00	30

Operand before Operation:

01111001

0030

Operand after Operation:

01111000

0030

SET BITS ON MASKED (SBN)

Op Code (hex)	Q-Byte[1]	Operand Address[2]	
Byte 1	Byte 2	Byte 3	Byte 4
3A	I	Operand 1 direct	
7A	I	Op 1 disp from XR1	
BA	I	Op 1 disp from XR2	

[1]The Q-byte contains a single-byte binary mask specifying operand bits to be turned on.
[2]Operand 1 is a single byte field; operand 2 is not used.

Operation

The system examines the Q-byte, bit by bit. Whenever it encounters a binary 1 in the Q-byte, it sets the corresponding bit in the operand byte to 1; whenever the system encounters a binary 0 in the Q-byte, it leaves the corresponding bit in the operand unchanged.

Condition Register

This instruction does not affect the condition register.

Example

Instruction:

3A	01011010	00	20

Operand before Operation:

00001100

Operand after Operation:

01011110

SENSE I/O (SNS)

Op Code (hex)	Q-Byte[1]	Operand Address[2]	
Byte 1	Byte 2	Byte 3	Byte 4
30	I	Operand 1 direct	
70	I	Op 1 disp from XR1	
B0	I	Op 1 disp from XR2	

[1]I (Q-Byte) = DA-code, M-code, and N-code, where:
DA (bits 0-3) = the address of the device being sensed
M (bit 4) = an address modifier bit (not always used)
N (bits 5-7) = a code that specifies the register or unit being sensed

[2]The operand is always a 2-byte field and is addressed by its rightmost position.

Operation

This operation stores 16 bits of data from the source specified by the N-code in the main storage location specified by the operand address. At the same time, when an I/O LSR is specified as the source on a Model 15, the CPU also moves the high-order bits from the LSR into the PMR. These are the I/O > 64K bit and, if the system is equipped with more than 128K or more than 256K of storage, the I/O > 128K bit or the I/O > 256K bit. The contents of the PMR must be stored before its contents are changed to retrieve these bits.

A Q-byte of 00 specifies that the data source is to be the ADDRESS/DATA switches on the system control panel. Specifications for other data sources are discussed with the appropriate I/O device sense I/O instruction.

Condition Register

This instruction does not affect the condition register.

Example

Instruction:

30	C0	05	FF

30 – Sense Command
C – 5445
0 – Drive 1, Bytes 0 and 1
05FF – Main Storage Address

Status Bytes at Disk before Operation:

81	00

81 – No-Op and Seek Check

Operand before Operation:

00	AB
05FE	05FF

No Meaning

Operand after Operation:

81	00
05FE	05FF

Status Bytes at Disk after Operation:

01	00

01 – Seek Check

START I/O (SIO)

Op Code (hex)	Q-Byte[1]	R-Byte[2]
Byte 1	Byte 2	Byte 3
F3	I2	I1

[1] I2 = DA-, M-, and N-codes where, usually:
DA (bits 0 3) = the address of the device performing a function
M (bit 4) = an address modifier bit (not always used)
N (bits 5-7) = a code that specifies the function being performed

[2] I1 = control code that specifies additional functions

Operation

The operation of start I/O for each individual device is discussed under that device.

Condition Register

This instruction does not affect the condition register.

Program Notes

- If a unit check condition that prevents the execution of the start I/O instruction exists in the addressed device, the processor usually treats the start I/O as a no-op instruction.
- Any unit check condition that does not prevent the execution of a start I/O instruction is reset by the start I/O instruction, and the instruction is executed. (*Exception:* This not does not apply to the 3340.)
- A start I/O instruction that specifies the reset of an op-end interrupt condition is executed regardless of any unit check condition in the addressed device.
- Start I/O is a privileged instruction on Model 15.
- A start I/O to a not-ready or busy device results in:
  - A program level advance to a system using the dual program feature, or
  - The program looping on the start I/O instruction until the device becomes ready or not busy on a system without an enabled dual program feature

Example

Instruction (to read key and data from drive 1):

F3	1100	0	001	00000000

Disk Drive Control Register:

02	00

Disk Drive Data Register:

04	00

Disk Drive Control Field:

F	C	C	H	H	R	K_L	D_L	D_L	N
00	00	14	00	06	10	00	00	28	00

0200 (CPU storage address)

The disk drive data field starts at main storage address 0400 and ends at 0428. Before the operation, the DDDF should be initialized to blanks; at the end of the operation, the DDDF contains data from cylinder 14, head 6, record 10 (hex).

STORE CPU (SCP)—MODEL 12C ONLY

Op Code (hex)	Q-Byte[1]	Operand Address[2]	
Byte 1	Byte 2	Byte 3	Byte 4
3E	Rx	Operand 1 direct	
7E	Rx	Op 1 disp from XR1	
BE	Rx	Op 1 disp from XR2	

[1] Rx = assigned code for register whose data is to be stored in the operand (Figure 2-5).
[2] Operand addressed is a 2-byte field.

Operation

This instruction stores the contents of the register or registers specified by the Q-byte in the storage location specified by the operand 1 address. The storage location specified is addressed by its low-order (rightmost) byte.

The Model 12C assembler does not recognize the mnemonic SCP, but the hardware recognizes the op code for this command.

To use this command for a Model 12C assembler, the programmer has the following options:

- Code the instruction using DCs
- Use $SCP macro (see *IBM System/3 Models 8, 10, and 12 System Control Programming Macros Reference Manual*, GC21-7562)

Program Notes, General

- Two bytes of data are transferred by the instruction.
- The storage location specified is addressed by its low-order (rightmost) byte.

Q-Byte in Hex	Information	Register Specified	
00		00 and 01	
01	Operand content:	02 and 03	ATT
02		04 and 05	
03	*Bits* *Function*	06 and 07	
04	0 Not used	08 and 09	
05	1 Not used	0A and 0B	ATT
06	2 Address >64K bit	0C and 0D	
07	3-7 Bits 0-4 of the high-order	0E and 0F	
08	byte of the address	10 and 11	
09		12 and 13	ATT
0A		14 and 15	
0B		16 and 17	
0C		18 and 19	
0D		1A and 1B	ATT
0E		1C and 1D	
0F		1E and 1F	
10	Program mode	Program level 1	
11		Program level 2	
18		Interrupt level 0	
19		Interrupt level 1	
1A		Interrupt level 2	
1B		Interrupt level 3	PMR
1C		Interrupt level 4	
40	Program mode	Current program mode (program mode register for the program or interrupt level that is active when the instruction is executed)	

Figure 2-5. Register ID Chart for SCP and LCP Q-Bytes

Program Notes, ATT Register

- The contents of the even-numbered address translate table (ATT) register is stored at the addressed storage position. The contents of the odd-numbered register is stored at the addressed position minus 1.

- The address translate table register bits have these assigned meanings:

Bit Position	Meaning
0-1	Not used
2	Address > 64K bit.
3-7	Bits 0-4 of the high-order (leftmost) byte of the address.

- Address translation does not occur for B-cycles of this instruction, regardless of the state of bit 1 (EB cycle address translate) of the program mode register.

- The contents of the ATT registers are not set to any value after a power up function, so the ATT registers must be initialized before activating the translation function in the program mode register.

Program Notes, Program Mode Register

- The instruction moves 2 bytes of data to storage. In the first byte:

  Bit 0 = Not used
  Bit 1 = EB cycle (operand 1) address translate
  Bit 2 = EA cycle (operand 2) address translate
  Bit 3 = I cycle (instruction) address translate
  Bit 4 = Not used
  Bit 5 = I/O cycle address translate
  Bit 6 = Not used
  Bit 7 = Mask interrupt

  The second byte is set to 0's.

- A bit set to 1 turns a function on; a bit set to 0 turns the associated function off.

- The mask interrupt bit is common to all program mode registers. Therefore, setting this bit on for one machine level (level 0 through 4, or either main program level) sets the mask interrupt bit for all levels to the same setting.

- The program mode register of any level may be changed at any time, but be cautious when changing the current PMR. If the current PMR is changed, the new values entered will be effective for the next instruction issued for that level. *Exception:* the mask interrupt bit becomes effective immediately, rather than when the next instruction is issued. This allows the instruction to inhibit any interrupt requests that might occur at operation end for the current instruction.

- The contents of the program mode registers are indeterminate after power up and must be initialized before using. To ensure proper system operation, initialize the program level first, then the interrupt level registers.

STORE CPU (SCP)—MODEL 15 ONLY

Op Code (hex)	Q-Byte[1]	Operand Address[2]	
Byte 1	Byte 2	Byte 3	Byte 4
3E	Rx	Operand 1 direct	
7E	Rx	Op 1 disp from XR1	
BE	Rx	Op 1 disp from XR2	

[1]Rx = assigned code for register whose data is to be stored in the operand (Figure 2-6).
[2]Operand addressed is a 2-byte field.

Operation

This instruction stores the contents of the register or registers specified by the Q-byte in the storage location specified by the operand 1 address. The storage location specified is addressed by its low-order (rightmost) byte.

Program Notes, General

- The SCP instruction can be executed in privileged mode only.
- Two bytes of data are transferred by the instruction.
- The storage location specified is addressed by its low-order (rightmost) byte.

Program Notes, Storage Protect Register on Models C and D

- The storage protect register stores 2 binary bits. These bits and their meanings are:

Bit Position	Meaning
0	Read and write storage protect key. Storage is protected from all read and write operations when this bit is 1.
1	Write storage protect key. Storage is protected from all write operations when this bit is 1.

- Powering down can alter contents of the storage protect registers. Therefore, the registers must be initialized after a power up function before they are used for a storage protect function or before an address translation function occurs.
- The content of the even-numbered storage protect register is stored at the addressed storage position. The content of the odd-numbered register is stored at the addressed position minus 1.
- The program mode register storage protect bit (bit 6) should be off when issuing a store-CPU instruction or load-CPU instruction specifying the storage protect registers.

Program Notes, ATT Register

- The contents of the even-numbered address translate table (ATT) register is stored at the addressed storage position. The contents of the odd-numbered register is stored at the addressed position minus 1.
- The address translate table register bits have these assigned meanings:

Bit(s)	Models A and B	Models C and D
0	Read and write storage protect key. Storage is protected from all read and write operations when this bit is 1.	Address > 256K bit (Models D25 and D26 only)
1	Write protect key. Storage is protected from all write operations when this bit is 1.	Address > 128K bit
2	Address > 64K bit.	Address > 64K bit
3-7	Bits 0-4 of the high-order (leftmost) byte of the address.	Bits 0-4 of the high-order (leftmost) byte of the address.

- Address translation does not occur for B-cycles of this instruction, regardless of the state of bit 1 (EB cycle address translate) of the program mode register.
- The program mode register storage protect bit (bit 6) should be off when issuing a store-CPU instruction (SCP) or load-CPU instruction (LCP) specifying the address translate table (ATT).
- The contents of the ATT registers are not set to any value after a power up function, so the ATT registers must be initialized before activating translation or storage protection functions in the program mode register.

Q-Byte in Hex	Information	Register Specified	
00 01 02 03	Operand content: *Bits* *Function* 0 Storage fetch protect bit 1 Storage write protect bit 2 Address >64K bit 3-7 Bits 0-4 of the high-order byte of the address	00 and 01 02 and 03 04 and 05 06 and 07	ATT (and storage protect register for Models C and D)
04 05 06 07		08 and 09 0A and 0B 0C and 0D 0E and 0F	ATT (and storage protect register for Models C and D)
08 09 0A 0B		10 and 11 12 and 13 14 and 15 16 and 17	ATT (and storage protect register for Models C and D)
0C 0D 0E 0F		18 and 19 1A and 1B 1C and 1D 1E and 1F	ATT (and storage protect register for Models C and D)
10 18 19 1A 1B 1C 1D 1E 1F	Program mode	Program level Interrupt level 0 Interrupt level 1 Interrupt level 2 Interrupt level 3 Interrupt level 4 Interrupt level 5 Interrupt level 6 Interrupt level 7	PMR
20 30	Program check	Check address register Check status register	
40	Program mode	Current program mode (program mode register for the program or interrupt level that is active when the instruction is executed)	
50 51 52 53	Operand content: *Bits* *Function* 0 Address >256K bit 1 Address >128K bit 2 Address >64K bit 3-7 Bits 1-4 of the high-order byte of the address (If an SCP instruction is issued on a Model A or B, the CPU stores hex 00 bytes in the addressed storage positions. If an LCP is issued to a Model A or B, the CPU does not alter the contents of the register.)	00 and 01 02 and 03 04 and 05 06 and 07	ATT (Models C and D)
54 55 56 57		08 and 09 0A and 0B 0C and 0D 0E and 0F	ATT (Models C and D)
58 59 5A 5B		10 and 11 12 and 13 14 and 15 16 and 17	ATT (Models C and D)
5C 5D 5E 5F		18 and 19 1A and 1B 1C and 1D 1E and 1F	ATT (Models C and D)
60 61 62 63	Operand content: *Bits* *Function* 0 Storage fetch protect bit 1 Storage write protect bit 2-7 Not used (If an SCP instruction is issued on a Model A or B, the CPU stores hex 00 bytes in the addressed storage positions. If an LCP is issued to a Model A or B, the CPU does not alter the contents of the register.)	00 and 01 02 and 03 04 and 05 06 and 07	Storage protect register (Models C and D)
64 65 66 67		08 and 09 0A and 0B 0C and 0D 0E and 0F	Storage protect register (Models C and D)
68 69 6A 6B		10 and 11 12 and 13 14 and 15 16 and 17	Storage protect register (Models C and D)
6C 6D 6E 6F		18 and 19 1A and 1B 1C and 1D 1E and 1F	Storage protect register (Models C and D)

Figure 2-6. Register ID Chart for SCP and LCP Q-Bytes

Program Notes, Program Check Register

- The program check registers store error definition data whenever a program check occurs, if (1) Interrupt level 7 is enabled, and (2) The CPU is not executing a program check routine when another program error occurs.

- The program check address register holds the 16 low-order bits of the physical location of the op code, the Q-code, or the operand address portion of the instruction being executed when the error occurs. The store CPU instruction stores byte 1 (which contains the low-order 8 bits of the address) in the location specified by the operand address. Byte 2 is stored in the operand address minus 1. The three high-order bits of the physical address are bits 6, 7, and 0 of byte 2 in the program check status register.

- The status register bits and their meanings are shown in Figure 2-7. Byte 1 (rightmost byte) of the status register is stored at the location specified by the operand address portion of the Store CPU instruction. Byte 2 is stored at the operand address minus 1.

- Resetting interrupt level 7 resets the program check registers.

- If program check status byte 1, bit 1 is on and bit 4 is off, the error was caused by an invalid Q-byte of an I/O instruction. The IAR of the erring program will contain the logical address of the I-Q byte plus 1 of the failing instruction.

- If program check status byte 1, bits 1 and 4 are both on, the error was caused by a privileged operation that was detected during the I-Q cycle. The instruction address registers (IAR) of the erring program will contain logical address of the I-Q byte plus 1 of the failing instruction.

- If program check status byte 1, bit 2 is on, the error was caused by an invalid op code. The IAR of the erring program will contain the logical address of the I-Q byte of the failing instruction.

- Errors indicated by program check status byte 1, bit 0 or 3 can occur during any CPU cycle, so the IAR contents are not meaningful.

Program Notes, Program Mode Register

- The instruction moves 2 bytes of data to storage. In the first byte:
  - Bit 0 = I/O > 128K
  - Bit 1 = EB cycle (operand 1) address translate
  - Bit 2 = EA cycle (operand 2) address translate
  - Bit 3 = I cycle (instruction) address translate
  - Bit 4 = Privileged mode
  - Bit 5 = I/O > 64K
  - Bit 6 = Storage protect
  - Bit 7 = Mask interrupt

  In the second byte:
  - Bits 0-6 = Set to 0's
  - Bit 7 = I/O > 256K

- A bit set to 1 turns a function on; a bit set to 0 turns the associated function off.

- The mask interrupt bit is common to all program mode registers. Therefore, setting this bit on for one machine level (level 0 through 7, or main program level) sets the mask interrupt bit for all levels to the same setting.

- The program mode register of any level may be changed at any time while in the privileged mode, but be cautious when changing the current PMR. If the current PMR is changed, the new values entered will be effective for the next instruction issued for that level. *Exception:* the mask interrupt bit becomes effective immediately, rather than when the next instruction is issued. This allows the instruction to inhibit any interrupt requests that might occur at operation end for the current instruction.

- The contents of the program mode registers are indeterminate after power up and must be initialized before using. To ensure proper system operation, initialize the program level first, then the interrupt level registers.

- Only this instruction can modify the I/O > 256K bit of the PMR.

Condition Register

This instruction does not affect the condition register.

Example

Instruction:

3E	18	00	20

Interrupt Level 0 Program Mode Register:

A8

Operand before Operation:

001F	0020
32	CD

Operand after Operation:

001F	0020
00	A8

Byte 1 Bit	Meaning When Set to 1	Byte 2 Bit	Meaning When Set to 1
0	Address violation	0	>256K address bit
1	Invalid Q-byte	1	Not used
2	Invalid op code	2	Binary value of 4[1]
3	Invalid address	3	Binary value of 2[1]
4	Privileged operation	4	Binary value of 1[1]
5	CE diagnostic bit	5	Any interrupt 0-7[2]
6	CE diagnostic bit	6	>64K address bit
7	CE diagnostic bit	7	>128K address bit

[1]The sum of these binary bits indicates the interrupt level active when the program check occurred.

[2]If bit 5 is on, but bit 2, 3, or 4 is not on, interrupt level 0 caused the stop. If bits 2, 3, 4, 5, and 6 are all off, the stop occurred while the CPU was operating in the basic program level.

Figure 2-7. Program Check Status Bits

STORE REGISTER (ST)

Op Code (hex) Byte 1	Q-Byte[1] Byte 2	Operand Address[2] Byte 3	Byte 4
34	Rx	Operand 1 direct	
74	Rx	Op 1 disp from XR1	
B4	Rx	Op 1 disp from XR2	

[1]Rx specifies the register whose contents are to be stored.
[2]Operand 1 is a 2-byte field addressed by its rightmost byte; operand 2 is not used.

Operation

This instruction places the contents of the register specified by the Q-byte into the 2-byte field specified by the operand address.

The Q-byte specifies the register to be stored. The high-order bit of the Q-byte, bit 0, specifies which of two groups of registers is to be addressed, and the low-order bit specifies which register within each group is to be stored.

If the high-order bit is 0, the selected group consists of the following seven local storage registers, each represented by a single bit:

Bit	Register
1	Program address recall register for Model 15; level 2 program instruction address register (P2IAR) for Models 8, 10, and 12
2	PIAR for Model 15; program level 1 instruction address register (P1IAR), for Models 8, 10, and 12
3	Current instruction address register in use when the store register instruction is executed[1]
4	Current address recall register[1]
5	Program status register; the high-order byte of this register is the length count recall register and has no program significance; the low-order byte is the image of the condition register
6	Index register 2 (Index register 2 for current program level for Models 8, 10, and 12)
7	Index register 1 (Index register 1 for current program level for Models 8, 10, and 12)

[1]May be one of the interrupt level registers or one of the program level registers.

If the high-order bit of the Q-code is 1, the interrupt instruction address registers are selected as follows:

Bit	Register	Interrupt Level	
None	IAR-0	Interrupt level 0	
1	IAR-1	Interrupt level 1	
2	IAR-2	Interrupt level 2	
3	IAR-3	Interrupt level 3	
4	IAR-4	Interrupt level 4	
5	IAR-5	Interrupt level 5	Model 15 only
6	IAR-6	Interrupt level 6	Model 15 only
7	IAR-7	Interrupt level 7	Model 15 only

Program Note

This instruction must not be used to store more than one register at a time. An attempt to execute this instruction when Q-bit 0 = 1 and the system is not in privileged mode will result in a program check interrupt or a processor check.

Condition Register

This instruction does not affect the condition register.

Example

Instruction:

34	00001000	B2	BB

Address Recall Register:

0A	CD

Operand before Operation:

B2BA	B2BB
2F	C2

Operand after Operation:

B2BA	B2BB
0A	CD

SUBTRACT LOGICAL CHARACTERS (SLC)

Op Code (hex)	Q-Byte[1]	Operand Addresses[2]			
Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
0F	L1-1	Operand 1 direct		Operand 2 direct	
1F	L1-1	Operand 1 direct		Op 2 disp from XR1	
2F	L1-1	Operand 1 direct		Op 2 disp from XR2	
4F	L1-1	Op 1 disp from XR1	Operand 2 direct		
5F	L1-1	Op 1 disp from XR1	Op 2 disp from XR1		
6F	L1-1	Op 1 disp from XR1	Op 2 disp from XR2		
8F	L1-1	Op 1 disp from XR2	Operand 2 direct		
9F	L1-1	Op 1 disp from XR2	Op 2 disp from XR1		
AF	L1-1	Op 1 disp from XR2	Op 2 disp from XR2		

[1] L1-1 = number of bytes in operand 1, minus 1
[2] Maximum length of an operand is 256 bytes; both operands must be the same length. The operands may overlap. Address operands by their rightmost bytes.

Operation

This instruction subtracts the binary number in operand 2 from the binary number in operand 1 and stores the result in operand 1. If the number stored in the second operand is larger than the number stored in the first operand, the answer develops as though the first operand has an additional high-order binary digit. The result can never be negative. For example:

First operand	0110 1101
Second operand	0111 1110
Result	1110 1111

Program Note

Overlapping the operands with the rightmost byte of the first operand to the left of the rightmost byte of the second operand destroys part of the second operand before it is used in the operation.

Condition Register

Bit	Name	Condition Indicated
7	Equal	Zero result
6	Low	Operand 1 smaller than operand 2
5	High	Operand 1 greater than operand 2
4	Decimal overflow	Bit not affected
3	Test false	Bit not affected
2	Binary overflow	Bit not affected

Example

Instruction:

AF	03	00	10

Index Register 2 = 0CC0

Operand 1 before Operation:

10010110	01011010	01110111	10111111
0CBD	0CBE	0CBF	0CC0

Operand 2:

01110100	10000110	01100010	10100100
0CCD	0CCE	0CCF	0CD0

Operand 1 after Operation:

00100001	11010100	00010101	00011011
0CBD	0CBE	0CBF	0CC0

Condition Register:

Before Operation:	After Operation:
00000000	00000100

SUBTRACT ZONED DECIMAL (SZ)

Op Code (hex)	Q-Byte[1]	Operand Addresses[2]			
Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
07	L1-L2 L2-1	Operand 1 direct		Operand 2 direct	
17	L1-L2 L2-1	Operand 1 direct		Op 2 disp from XR1	
27	L1-L2 L2-1	Operand 1 direct		Op 2 disp from XR2	
47	L1-L2 L2-1	Op 1 disp from XR1	Operand 2 direct		
57	L1-L2 L2-1	Op 1 disp from XR1	Op 2 disp from XR1		
67	L1-L2 L2-1	Op 1 disp from XR1	Op 2 disp from XR2		
87	L1-L2 L2-1	Op 1 disp from XR2	Operand 2 direct		
97	L1-L2 L2-1	Op 1 disp from XR2	Op 2 disp from XR1		
A7	L1-L2 L2-1	Op 1 disp from XR2	Op 2 disp from XR2		

[1]L1-L2 (4 bits) = number of bytes in operand 1, minus the number of bytes in operand 2
L2-1 (4 bits) = number of bytes in operand 2, minus 1
Maximum length of operand 1 is 31 bytes; maximum length of operand 2 is 16 bytes.
[2]The operands may overlap. Address operands by their rightmost bytes.

Operation

This instruction algebraically subtracts operand 2 from operand 1, byte by byte, and stores the result in operand 1. The processing unit sets the zone bits of all operand 1 bytes except the rightmost byte to hex F (binary 1111). It sets the zone bits of the rightmost byte in operand 1 to (1) hex F if the result of the operation is either positive or 0, or (2) hex D (binary 1101) if the result is negative.

Program Notes

- The second operand remains unchanged unless the fields overlap.
- If operands are overlapped and the operand 1 address is lower than the operand 2 address, data in the overlapped positions of operand 2 is destroyed before it is used in the operation.
- The system does not check for valid decimal digits in either operand.
- The decimal-overflow-condition indication, which may be set during this operation, is reset by:
  - A system reset
  - Testing decimal overflow with a branch-on-condition or jump-on-condition instruction
  - Loading a 0 in bit 4 of the program status register using the load-register instruction
- The system saves the starting address of operand 1 in the address recall register.

Condition Register

Bit	Name	Condition Indicated
7	Equal	Zero result
6	Low	Negative result
5	High	Positive result
4	Decimal overflow	Carry occurred from the leftmost position of operand 1
3	Test false	Bit not affected
2	Binary overflow	Bit not affected

Example

Instruction:

07	22	00	10	00	20

Operand 1 before Operation:

F7	F6	F3	F6	F9
000C	000D	000E	000F	0010

Operand 2:

F4	F2	F5
001E	001F	0020

Operand 1 after Operation:

F7	F5	F9	F4	F4
000C	000D	000E	000F	0010

Condition Register:

Before Operation:	After Operation:
00000000	00000100

TEST BITS OFF MASKED (TBF)

Op Code (hex)	Q-Byte[1]	Operand Address[2]	
Byte 1	Byte 2	Byte 3	Byte 4
39	I	Operand 1 direct	
79	I	Op 1 disp from XR1	
B9	I	Op 1 disp from XR2	

[1]The Q-byte contains a single-byte binary mask specifying operand bits for testing.
[2]Operand 1 is a single-byte field; operand 2 is not used.

Operation

This instruction tests specified bits in the operand byte for a binary 1. For each mask bit (Q-byte bit) that is a 1, the system tests the corresponding bit in the operand. If any tested bit is a 1, the system turns the test false indicator (in the condition register) on.

Program Notes

- The operand and Q-byte remain unchanged.
- Test false condition is turned off by system reset, using test false as a condition in a branch-on-condition or jump-on-condition instruction, or by loading a binary 0 into condition register bit 11 (bit 3 of the rightmost condition register byte).

Condition Register

Bit	Name	Condition Indicated
7	Equal	Bit not affected
6	Low	Bit not affected
5	High	Bit not affected
4	Decimal overflow	Bit not affected
3	Test false	One of tested bits on
2	Binary overflow	Bit not affected

Example

Instruction:

39	01101100	00	25

Storage Operand:

10010100

0025

Condition Register after Operation:

00010000

TEST BITS ON MASKED (TBN)

Op Code (hex)	Q-Byte[1]	Operand Address[2]	
Byte 1	Byte 2	Byte 3	Byte 4
38	I	Operand 1 direct	
78	I	Op 1 disp from XR1	
B8	I	Op 1 disp from XR2	

[1]The Q-byte contains a single-byte binary mask specifying operand bits for testing.
[2]Operand 1 is a single-byte field; operand 2 is not used.

Operation

This instruction tests specified bits in the operand byte for an on state. For each mask bit (Q-byte bit) on, the system tests the corresponding bit in the operand. If any tested bit is off, the system turns the test false indicator (in the condition register) on.

Program Notes

- The operand and Q-byte remain unchanged.
- Test false condition is turned off by:
  - System reset
  - Using test false as a condition in a branch-on-condition or a jump-on-condition instruction
  - Loading a binary 0 into condition register bit 3 (bit 3 of the rightmost program status register byte) using a load-register instruction.

Condition Register

Bit	Name	Condition Indicated
7	Equal	Bit not affected
6	Low	Bit not affected
5	High	Bit not affected
4	Decimal overflow	Bit not affected
3	Test false	One of the tested bits not on
2	Binary overflow	Bit not affected

Example

Instruction:

38	00010110	00	21

Storage Operand

10010101

0021

Condition Register after Operation:

00010000

TEST I/O AND BRANCH (TIO)

Op Code (hex)	Q-Byte[1]	Operand Address[2]	
Byte 1	Byte 2	Byte 3	Byte 4
C1	I	Operand 2 direct	
D1	I	Op 2 disp from XR1	
E1	I	Op 2 disp from XR2	

[1]I (Q-Byte = DA-code, M-code, and N-code where,
DA (bits 0-3) = address of device tested
M (bit 4) = address modifier
N (bits 5-7) = a code that specifies the condition for which the device is tested.

[2]The operand address specifies the branch-to-address.

Operation

This operation tests the condition specified by the Q-byte for the addressed device. If the condition is present, the branch-to address is placed in the instruction address register and the next sequential instruction address is placed in the address recall register. If the condition is not present, the next sequential address is used and the branch-to address is placed in the address recall register. The address placed in the address recall register remains there until the next decimal-add, decimal-subtract, insert-and-test-characters, or branch instruction is executed.

Condition Register

This instruction does not affect the condition register.

Example

Instruction:

5445
Drive 2 Not Ready or Unit Check

C1	C8	02	00
0100	0101	0102	0103
Test I/O		Branch to Address	

Status Byte 1:

10000000	Bit 0 on indicates a disk drive error condition.

Instruction Address Register before Operation:

01	04	Next Sequential Instruction

Address Recall Register before Operation:

02	00	Branch-to Address (loaded when instruction is taken from main storage)

Instruction Address Register after Operation:

02	00	Branch-to Address

Address Recall Register after Operation:

01	04	Address of Return Point in Main Program

Contents of Registers Were Swapped Because Branch Occurred

ZERO AND ADD ZONED (ZAZ)

Op Code (hex) Byte 1	Q-Byte[1] Byte 2	Operand Addresses[2] Byte 3	Byte 4	Byte 5	Byte 6
04	L1-L2 | L2-1	Operand 1 direct		Operand 2 direct	
14	L1-L2 | L2-1	Operand 1 direct		Op 2 disp from XR1	
24	L1-L2 | L2-1	Operand 1 direct		Op 2 disp from XR2	
44	L1-L2 | L2-1	Op 1 disp from XR1	Operand 2 direct		
54	L1-L2 | L2-1	Op 1 disp from XR1	Op 2 disp from XR1		
64	L1-L2 | L2-1	Op 1 disp from XR1	Op 2 disp from XR2		
84	L1-L2 | L2-1	Op 1 disp from XR2	Operand 2 direct		
94	L1-L2 | L2-1	Op 1 disp from XR2	Op 2 disp from XR1		
A4	L1-L2 | L2-1	Op 1 disp from XR2	Op 2 disp from XR2		

[1] L1-L2 (4 bits) = number of bytes in operand 1, minus the number of bytes in operand 2
L2-1 (4 bits) = number of bytes in operand 2, minus 1
Maximum length of operand 1 is 31 bytes; maximum length of operand 2 is 16 bytes.
[2] The operands may overlap. Address operands by their rightmost bytes.

Operation

This instruction moves data from the second operand, byte by byte starting with the rightmost byte, into the first operand. If the first operand is longer than the second operand, the processing unit fills the unused high-order positions with decimal 0's (hex F0).

The processing unit sets the zone bits of all bytes except the rightmost byte in the first operand to hex F (binary 1111). It sets the zone bits of the rightmost byte in the first operand to (1) hex F if the value transferred is either zero or positive, or (2) hex D (binary 1101) if the value transferred is negative.

Program Notes

- The second operand remains unchanged unless the fields overlap.
- If operands are overlapped and the operand 1 address is lower than the operand 2 address, data in the overlapped positions of operand 2 is destroyed before it is used in the operation.

Condition Register

Bit	Name	Condition Indicated
7	Equal	Zero result
6	Low	Negative result
5	High	Positive result
4	Decimal overflow	Bit not affected
3	Test false	Bit not affected
2	Binary overflow	Bit not affected

Example

Instruction:

04	22	00	10	00	20

Operand 1 before Operation:

F7	F6	F3	F6	F9
000C	000D	000E	000F	0010

Operand 2:

F4	F2	F5
001E	001F	0020

Operand 1 after Operation:

F0	F0	F4	F2	F5
000C	000D	000E	000F	0010

Condition Register:

Before Operation:	After Operation:
00000000	00000100

Figures 3-1, 3-2, and 3-3 show the data flow for the basic processing unit. Input data enters the A-register, passes through the ALU (arithmetic and logical unit), then enters storage. When in dual-byte mode (Models 12 and 15 only), bytes of data being sent to odd-numbered storage locations bypass both the A-register and the ALU; bytes of data being sent to the I/O unit from odd-numbered storage locations bypass both the B-register and the ALU. Data is taken from storage to the B-register. From the B-register, data enters the ALU to be operated on and directed to one of the following units:

- Op code register
- Q-register
- Condition register
- One of the local storage registers (LSRs)
- An I/O unit
- ATT register (Models 12C and 15)
- PMR register (Models 12C and 15)
- Storage protect register (Model 15 only)
- Main storage

REGISTERS

A-Register—All Models

The A-register temporarily stores 1 byte of data at a time before the data is processed by the ALU. Data can enter the A-register (via the ALU) from the B-register, from one of the local storage registers, or from an input device. When a Model 12 or 15 is operating in dual byte mode (transferring 2 bytes of data at a time between the disk attachment and main storage), odd-numbered storage position bytes do not pass through the A-register.

The A-register cannot be accessed by the program, but its contents can be displayed on the display panel.

Address Recall Register (ARR)—All Models

Each level (program level or interrupt level) on a System/3 has its own address recall register. The CPU uses the contents of the ARR during the various phases of an instruction.

During the instruction phase of a branch operation, the CPU fetches the branch-to address from the operand address portion of the instruction and loads it into the address recall register. During the execution phase of the operation, the CPU swaps the contents of the address recall register with the contents of the instruction address register if a branch is to occur. (If a branch is not to occur, the contents of the two registers remain unaltered.)

During the instruction phase of a decimal instruction, the CPU stores the operand 1 address in the address recall register.

During the execution phase of an insert-and-test-characters instruction, the CPU stores the address of the first significant digit encountered.

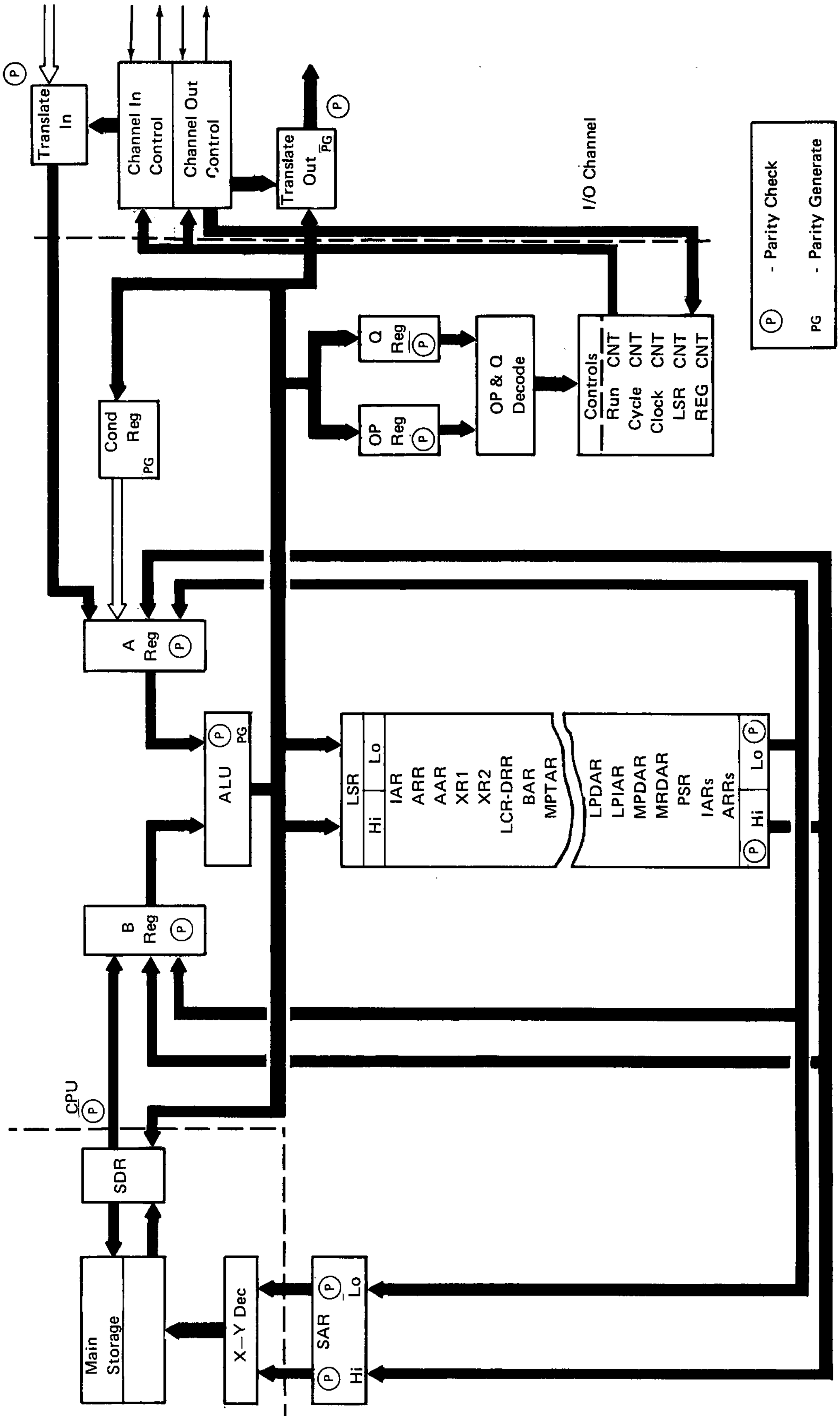
Arithmetic and Logical Unit (ALU)

The ALU performs all the arithmetic and logical functions for the processing unit. Data that is to be moved from any unit in the data flow to any other unit in the data flow (except the storage address register and A- and B-registers) usually passes through the ALU. In dual-byte mode, bytes associated with odd-numbered storage locations do not pass through the A-register, B-register, or the ALU. The ALU accepts 2 bytes of input and produces 1 byte of output.

The ALU output and state of the ALU can be displayed on the display panel.

B-Register—All Models

The B-register temporarily stores each data byte and instruction byte moved from storage to the ALU. This register cannot be accessed by the program, but its contents can be displayed by the display panel.



Note: The Model 12 has a separate path (not shown on this figure) for bytes between the disk attachment and odd-numbered main storage positions. Bytes move between even-numbered main storage positions and the attachment via the usual data path.

Figure 3-1. Processing Unit Data Flow, Models 8, 10, and 12

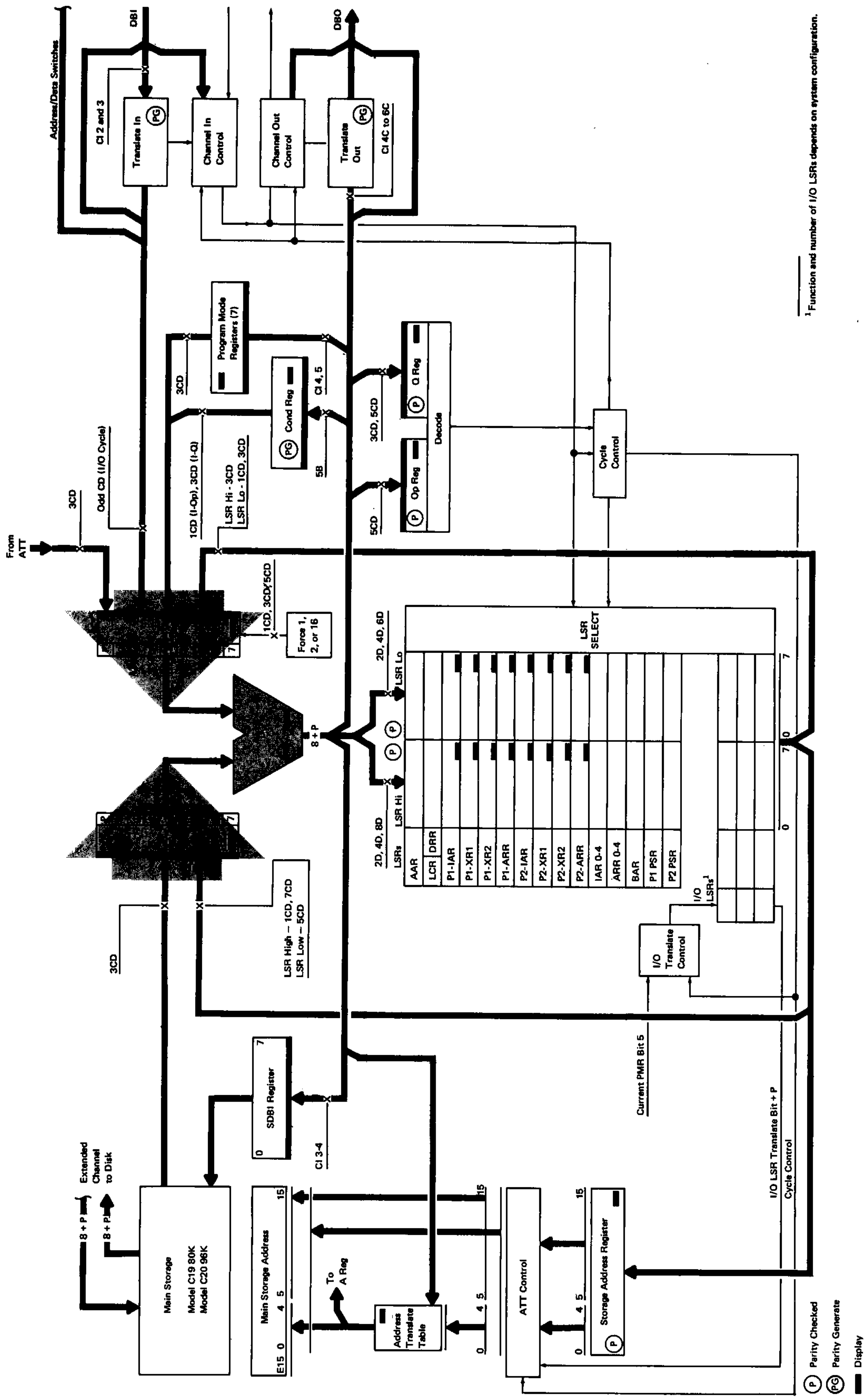


Figure 3-2. Processing Unit Data Flow, Model 12C

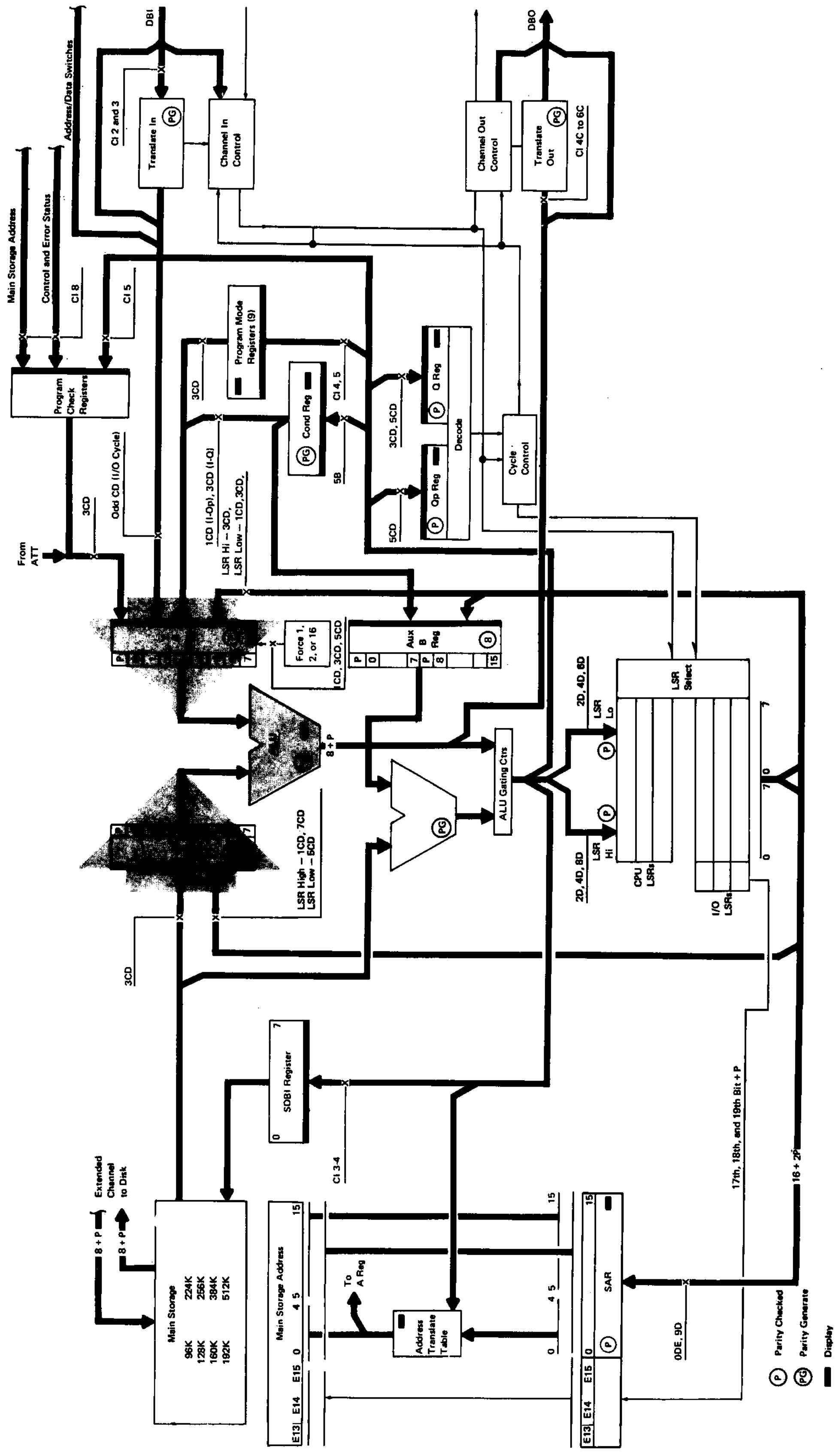


Figure 3-3. Processing Unit Data Flow, Model 15

Condition Register—All Models

Bits 2 through 7 of the condition register have assigned meanings which are set and reset during the operations shown in Figure 3-4. For example, bits in the condition register can indicate a high, a low, or an equal condition after a compare operation; after an arithmetic operation, bits can indicate that a binary or decimal overflow has occurred. The program can test this register for these conditions. The bits in this register are not reset by the same operation that set them. Instead, they act like six individual latch-type switches.

Bit	Bit Name	Operation That Sets Bit	Operation That Resets Bit
7 6 5	Equal Low High	• Zero and add zoned • Add zoned decimal • Subtract zoned decimal • Add logical characters • Subtract logical characters • Add to register • Edit • Load register that loads the associated bit in the program status register	• System reset • Load register that loads 0 into the associated bit position of the program status register • Instruction phase of one of the following: – Zero and add zoned – Add zoned decimal – Subtract zoned decimal – Add logical characters – Subtract logical characters – Add to register – Edit
4	Decimal overflow	• Add zoned decimal • Subtract zoned decimal • Load register that loads a 1 into bit position 4 of the program status register	• Instruction phase of next branch-on-condition instruction or jump-on-condition instruction that specifies test decimal overflow as a branch or jump condition. • Load register that loads 0 bit into bit position 4 of the program status register • System reset
3	Test false	• Test bits on • Test bits off • Load register that loads a 1 into bit position 3 of the program status register	• System reset • Instruction phase of next branch-on-condition instruction or jump-on-condition instruction that specifies test false as a branch or jump condition
2	Binary overflow	• Add logical characters • Load register that loads a 1 into bit position 2 of the program status register	• System reset • Instruction phase of next add-logical-characters instruction • Load register that loads a 0 into bit position 2 of the program status register
1 0	These bits are not used.		

Each program level on a system equipped with the Dual Program Feature has a condition recall register. This condition recall register holds data regarding the inactive level, retaining that data until the level again becomes active.

To understand the relationships between each of the condition recall registers and the condition register, and how the registers function, assume the following:

- The program levels are designated level A and level B, and are controlled by programs A and B, respectively.
- The condition recall register for level A is designated condition recall register A; the condition recall register for level B is designated condition recall register B.
- Level A is active at the start of the operation.

Assume program A issues an APL instruction to test a condition, and the tested condition exists. The CPU stores the contents of the condition register (data associated with program A) in condition recall register A, then loads the contents of condition recall register B into the condition register. This places conditions associated with program B in the condition register. When the next program level advance occurs, the CPU stores the contents of the condition register (now program B condition data) in condition recall register B and loads the contents of condition recall register A back into the condition register. In this manner the condition register always reflects conditions associated with the currently active program.

Whenever the program issues branch-on-condition and jump-on-condition instructions, bits stored in the condition register are checked. The program can store the contents of the condition register by issuing a store-register instruction; it can load information into the condition register by issuing a load-register instruction.

Unlike operations using the dual program mode, the program *must* store the contents of the condition register whenever entering an interrupt routine, and *must* reload the condition register at the end of the routine.

Data Recall Register (DRR)—All Models

The data recall register, which is a 1-byte register, shares a 2-byte local storage register with the 1-byte length count register. The processing unit uses the data recall register for two types of instructions:

- During the instruction cycles for single-address instructions, the processing unit stores the Q-byte in the data recall register.
- During each EA cycle for a two-address instruction, the processing unit places the byte of data read from operand 2 in the data recall register. Then, during the subsequent EB cycle, the processing unit moves this byte of data from the data recall register to the A-register as it moves the associated byte from operand 1 to the B-register. Then the processing unit moves both bytes from the A- and B-registers to the ALU for processing.

The program cannot access the data recall register, and its contents cannot be displayed.

Index Registers (XR1 and XR2)—All Models

Index register (XR1) and index register (XR2) are standard on all systems. Each Model 8, 10, or 12 that is equipped with the Dual Program Feature has a second pair of index registers, identical to the first pair, that support program level 2.

Each index register holds 2 bytes (16 bits). During base displacement addressing, the program must store the base address in the index register to be used to address the operand. Either register can be used for either operand, or the same register can be used for both operands.

The load-register, add-to-register, store-register, and load-address instructions can address the index registers. Operations which alter base addresses do not change the contents of the index register.

Instruction Address Register (IAR)—All Models

Each program level and interrupt level on a System/3 has its own instruction address register. The CPU uses the contents of instruction address registers to determine the address of the next byte of an instruction that is to be moved from main storage to one of the CPU registers.

During the instruction phase of an instruction, the CPU fetches the first byte of the instruction from the main storage position addressed by the IAR, then adds 1 to the address stored in the IAR. The CPU then repeats this fetch-byte-and-update-address process until all the bytes in the instruction have been stored in appropriate registers in the CPU. At the end of the instruction phase of an instruction, the instruction address register contains the address of the first byte of the next sequential instruction.

During the instruction phase of a branch instruction, the CPU fetches the branch-to address from the instruction and loads it into the address recall register. During the execution phase of the instruction, the CPU swaps the contents of the address recall register with the contents of the instruction address register if a branch is to occur. (If a branch is not to occur, the contents of the two registers are not altered.)

Pressing either the SYSTEM RESET key or the PROGRAM LOAD key resets the program level IAR (Model 15 only) and the program 1 IAR (Models 8, 10, and 12) to 0, but does not alter the content of any other IAR. However, the address in the program level IAR (Model 15) or the program 1 IAR (Models 8, 10, and 12) may have been traded for the address in the program level address recall register (Model 15) or the program 1 address recall register (Models 8, 10, and 12). Therefore, do not assume that the addresses in these registers remain constant throughout the system reset or program load operation.

Length Count Register (LCR)—All Models

The length count register, which is a 1-byte register, shares a 2-byte local storage register with the data recall register.

During instruction cycles of each two-address instruction, the CPU loads the number of bytes specified for each operand into the length count register (as the CPU stores the Q-byte in the Q-register). Then, after the CPU has processed each pair of associated bytes (one from each operand), it subtracts 1 from the length count register. Instruction execution continues until the length count register contains a zero value.

The program cannot access the length count register, and its contents cannot be displayed.

Local Storage Registers (LSRs)—All Models

The local storage registers hold data and addresses required for the execution of instructions. Each of the LSRs except the length count register, the length count recall register, the data recall register, and the program status register are 2-byte registers.

Pressing the system reset key or program load key resets the following registers to 0:

Models 8, 10, and 12	Model 15
Program level 1 instruction address register	Program level instruction address register
Program level 1 program status register	Program status register
Program level 2 program status register	

Pressing the PROGRAM LOAD key also resets to 0 the data address register for the device used for the program load function. The program must initialize all other instruction address registers, index registers, and I/O local storage registers before their use.

Fetching the first instruction from storage sets the program status register to condition equal. After the execution of the first instruction, the program status register (program level 1 program status register on Models 8, 10, and 12) remains at condition equal unless the instruction itself causes some condition other than equal to exist. In that case, the CPU sets the program status register to the resulting condition.

Processing Unit Local Storage Registers

System/3 CPU local storage registers and their acronyms are listed below. They are described in detail in separate writeups.

LSR Name	*Acronym*	*Note*
Length Count Register	LCR	1
Length Count Recall Register	LCRR	
Data Recall Register	DRR	1
Operand 1 (B Field) Address Register	BAR	
Operand 2 (A Field) Address Register	AAR	
Index Register 1	XR1	
Index Register 2	XR2	
Instruction Address Registers (IARs):		
Program Level 1 IAR	P1 IAR	2
Program Level 2 IAR	P2 IAR	2
Program Level IAR	PIAR	
Interrupt Level 0 IAR	IAR-0	
Interrupt Level 1 IAR	IAR-1	
Interrupt Level 2 IAR	IAR-2	
Interrupt Level 3 IAR	IAR-3	
Interrupt Level 4 IAR	IAR-4	
Interrupt Level 5 IAR	IAR-5	3
Interrupt Level 6 IAR	IAR-6	3
Interrupt Level 7 IAR	IAR-7	3
Address Recall Registers (ARRs):		
Address Recall Register	ARR	
Program Level ARR	PARR	
Interrupt Level 0 ARR	ARR-0	
Interrupt Level 1 ARR	ARR-1	
Interrupt Level 2 ARR	ARR-2	
Interrupt Level 3 ARR	ARR-3	
Interrupt Level 4 ARR	ARR-4	
Interrupt Level 5 ARR	ARR-5	3
Interrupt Level 6 ARR	ARR-6	3
Interrupt Level 7 ARR	ARR-7	3
Program Status Register	PSR	4

Notes:
1. The length count register and data recall register are each single-byte registers that share a 2-byte physical register.
2. Not used on Model 15.
3. Used only on Model 15.
4. The low-order (rightmost) byte of the program status register is used as the condition recall register (CRR). The high-order (leftmost) byte is used as the length count recall register (LCRR).

Input/Output Unit Local Storage Registers

Each I/O device has an adapter or an attachment feature, and each adapter except the MLTA and those for the 5471 and 5475 has at least one directly associated data address register (DAR). To service some adapters, the CPU has additional local storage registers used by the adapter for the I/O operation. For example, line printers must access specified areas of storage to examine the character set image, so the line printer adapter has a line printer image address register that directs the processing unit to the character set image field. The 5424 is an example of an I/O device that has more than one data address register: it has a print DAR, a punch DAR, and a read DAR.

The program must load the address of the applicable main storage field into each address register before the register is used for an I/O operation. During the operation the CPU updates the register, under control of the adapter, to reflect the next storage position to be accessed.

All LSR registers for Models 8, 10, and 12B have 16 bits that allow addressing hex storage positions 0000 through FFFF. For Model 12C, an additional (17th) bit is needed to address additional memory. This bit is the I/O cycle translate bit. For Model 15, additional bits are needed to address more than 64K of storage. Therefore, Models 15A and 15B LSRs provide an extra (17th) bit, which is the address greater than 64K bit. Models 15C and 15D provide both the 17th bit and an 18th bit which is the address greater than 128K bit. Model 15 D25 and Model 15 D26 provide an additional bit, which is the address greater than 256K bit.

I/O unit LSRs are discussed with the description of programming for the devices for which they are provided.

Op Code Register—All Models

At the start of the instruction phase of an operation, the CPU fetches a byte from the storage position specified by the instruction address register and places it in the op code register. This byte must be the first byte in an instruction, which is always the op code. Then the CPU examines the bits in the op code register to determine the operation to be performed and the number of bytes still to be fetched from storage and stored in registers prior to instruction execution. After establishing the appropriate circuits to complete the instruction phase of the operation, the CPU moves the remaining bytes of the instruction from storage into the appropriate registers.

Operand 1 Address Register (BAR)—All Models

During the instruction phase of operations using operand 1, the CPU stores the starting address of operand 1 in the operand 1 address register. During execution of the instruction, the CPU fetches the byte addressed by the operand 1 address register, then updates the address register to point at the next byte of operand 1 to be fetched. After the byte has been processed, the CPU fetches the next byte and updates the operand 1 address register. This sequence continues until the operation ends.

Operand 1 is addressed by its rightmost byte for all operations except insert and test characters. Therefore, to update the operand 1 address register during execution of the instruction, the CPU subtracts 1 from the stored address in the register. At the end of these operations, the operand 1 address register contains the address of the leftmost byte in operand 1 minus 1.

Operand 1 is addressed by its leftmost byte for insert and test characters operations. During these operations the CPU adds 1 to the address stored in the operand 1 address register to update the register for fetching the next byte to be processed. At the end of insert and test characters operations, the operand 1 address register contains the address of the rightmost byte of operand 1 plus 1.

Operand 1 is sometimes called the B-field, which is why the operand 1 address register is called the BAR.

The operand 1 address register cannot be addressed by the program and its contents cannot be displayed.

Operand 2 Address Register (AAR)—All Models

During the instruction phase of operations using operand 2, the CPU stores the starting address of operand 2 in the operand 2 address register. During execution of the instruction, the CPU fetches the byte addressed by the operand 2 address register and places it in the B-register, then updates the address register to point to the next byte of operand 2 to be fetched. After the byte in the B-register has been processed, the CPU fetches the next byte and updates the operand 2 address register. This sequence continues until the operation ends.

Operand 2 is addressed by its rightmost byte for all operations except insert and test characters. Therefore, to update the operand 2 address register during execution of the instruction, the CPU subtracts 1 from the stored address in the register. At the end of these operations, the operand 2 address register contains the address of the leftmost byte in operand 2 minus 1.

Operand 2 is addressed by its leftmost byte for insert and test characters operations. During these operations the CPU adds 1 to the address stored in the operand 2 address register to update the register for fetching the next byte to be processed. At the end of insert and test characters operations, the operand 2 address register contains the address of the rightmost byte of operand 2 plus 1.

Operand 2 is sometimes called the A-field, which is why this register is called the AAR.

The operand 2 address register cannot be addressed by the program, and its contents cannot be displayed.

Program Check Status Register and Program Check Address Register—Model 15 Only

These 16-position registers are both active while the program check interrupt (level 7) is enabled unless the CPU is executing a program check interrupt routine. If the 5415 detects an invalid address, an invalid Q-byte, an invalid operation code, a privileged operation in nonprivileged mode, or an address violation check, the 5415:

1. Sets bits in the program check status register to indicate what caused the check condition, to specify which 64K segment of storage the program was using when the check occurred, and to identify the program level that was active when the check occurred.

2. Stores the contents of the physical address in the program check address register.

3. Issues a program check interrupt request. This is interrupt level 7, the highest interrupt level, so the CPU must handle this interrupt before any others.

The program check registers are both reset by resetting interrupt level 7 by means of a command CPU instruction.

Program Mode Register (PMR)—Models 12C and 15

The program level and each interrupt level has an associated program mode register. A PMR bit controls whether the Models 12C and 15 perform each of the functions specified in the chart below during execution of the program assigned to its level:

PMR Function Activated by Binary 1

Byte 1 Bit	(High Order Address) Model 12C	Model 15
0	Not used	I/O > 128K (131,072 bytes, Models 15C and 15D only)
1	Address translate during EB (operand 1) cycles	Address translate during EB (operand 1) cycles
2	Address translate during EA (operand 2) cycles	Address translate during EA (operand 2) cycles
3	Address translate during I (instruction) cycles	Address translate during I (instruction) cycles
4	Not used	Privileged mode
5	Address translate during I/O (input/output) cycles	I/O > 64K (65,536 bytes)
6	Not used	Storage protect
7	Mask interrupts	Mask interrupts

PMR Function Activated by Binary 1

Byte 2 Bit	(Low Order Address) Model 12C	Model 15
0-6	Not used	Zeros (Reserved)
7	Not used	I/O > 256K (262,144 bytes)

Because each machine level has its own PMR, the interrupt routines and the mainline program can independently use or ignore the function associated with each bit. As interrupts occur asynchronously to take control away from the mainline program or lower priority interrupts, the 5415 automatically selects the appropriate PMR to control the CPU. A description of each program mode register bit follows.

Byte 1 Bit 0, I/O Greater Than 128K (Models 15C and 15D only)

This bit represents the 128K-bit (the bit representing a 131,072 decimal value) in a binary address. The CPU uses this bit while executing LIO and SNS instructions.

- If the I/O > 128K bit is on when the program issues a load I/O instruction to an associated I/O LSR, the I/O > 128K bit in the selected LSR turns on if the I/O > 128K bit is off when the program issues a load I/O instruction to the LSR, the I/O > 128K bit in the selected LSR turns off.

- If the I/O > 128K bit in the LSR for a sensed I/O device is on when the program senses the LSR, the system sets the program mode register I/O > 128K bit on if the I/O > 128K bit in the LSR for the selected I/O device is off when the program senses the LSR, the system sets the program mode register I/O > 128K bit off.

Because the system uses a two-instruction sequence (LCP, then LIO) to set the I/O > 128K bit in the LSR for the selected device, and a two-instruction sequence (SNS, then SCP) to inspect the I/O > 128K bit in the LSR for the selected I/O device, an interrupt may occur between the two instructions in either operation. However, it is not necessary to mask interrupts during these operations, as each interrupt level has its own program mode register and, therefore, cannot destroy the contents of another interrupt level PMR.

Byte 1 Bit 1, Address Translate EB Cycles

When this bit is on, the CPU uses the output of the address translate table (ATT) to develop a physical address during EB (operand 1 execute) cycles. If operand 1 resides at a storage address greater than 65,535, this bit must be on and the address translate table must be appropriately loaded. If operand 1 resides at an address of 65,535 or less, the ATT may or may not be used, as the programmer chooses. Bit 1 must not be set on until the ATT contents are set after power on. Furthermore, this bit should be off whenever the ATT contents are changed.

Byte 1 Bit 2, Address Translate EA Cycles

When this bit is on, the CPU uses the output of the address translate table (ATT) to develop a physical address during EA (operand 2 execute) cycles. If operand 2 resides at a storage address greater than 65,535, this bit must be on and the ATT must be appropriately loaded. If operand 2 resides at an address of 65,535 or less, the ATT may or may not be used, as the programmer chooses. Bit 2 must not be set on until the ATT contents are set after power on.

Byte 1 Bit 3, Address Translate I-Cycles

When this bit is on, the CPU uses the output of the address translate table (ATT) to develop a physical address during I (instruction) cycles. The CPU fetches the instruction to be executed from main storage during I cycles. If the instruction resides at a storage address greater than 65,535, this bit must be on and the ATT must be appropriately loaded. If the instruction resides at an address of 65,535 or less, the ATT may or may not be used as the programmer chooses. This bit must not be set until the address translate table contents are set after a power on.

Byte 1 Bit 4, Privileged Mode (Model 15 Only)

When this bit is on, the system operates in privileged mode and executes all of the instructions in the Model 15 instruction set. If this bit is off, the CPU cannot execute any of these instructions:

- Load I/O
- Sense
- Start I/O
- Test I/O
- Advance program level
- Halt program level
- Command CPU (except supervisor call, which is allowed)
- Load CPU
- Store CPU
- Add to register
- Load register
- Store register

(except Q-bytes 01, 02, 04, 08, 10, 20, and 40, which are allowed) [applies to Add to register, Load register, Store register]

A program check interrupt (or processor check, if interrupt level 7 is not enabled) occurs if the program issues any of the preceding instructions while the CPU is not in privileged mode.

Bit 4 has no significance for interrupt level 0. The system is automatically privileged whenever the CPU is in interrupt level 0, regardless of the state of the privileged mode bit in the interrupt 0 program mode register (PMR). This means that all instructions can be executed when the CPU is in interrupt level 0.

Byte 1 Bit 5, Address Translation I/O Cycles (Model 12C only)

This bit is used in conjunction with LIO and SNS commands to load and sense the I/O cycle translate bit associated with each I/O device LSR.

If this bit is on and load I/O instruction is issued to a device LSR, the I/O translate bit in the selected LSR will be set on. If this bit is off and load I/O instruction is issued to a device LSR, the I/O translate bit in the selected LSR will be set off. When a sense I/O command is issued to a device LSR, the PMR I/O address translate bit will be set to the state of the I/O translate bit in the selected LSR. The PMR may be stored in memory and bit 5 inspected to determine the contents of the device LSR. Since a two-instruction sequence (LCP, then LIO, or SNS, then SCP) is required to set or inspect the I/O translate bit in the I/O LSR, an interrupt may occur between these instructions. Each interrupt level, however, has its own PMR and one level will not destroy the results of another. It is not necessary to set the mask interrupt bit on while executing this two-instruction sequence.

Byte 1 Bit 5, I/O Greater Than 64K (Model 15 only)

This bit represents the 64K-bit (the bit representing a 65,536 decimal value) in a binary address. The CPU uses this bit while executing LIO and SNS instructions.

- If the I/O > 64K bit is on when the program issues a load I/O instruction to an associated I/O LSR, the address > 64K bit in the selected LSR turns on.

- If the I/O > 64K bit is off when the program issues a load I/O instruction to the LSR, the address > 64K bit in the selected LSR turns off.

- If the address > 64K bit in the LSR for a sensed I/O device is on when the program senses the LSR, the system sets the program mode register I/O > 64K bit on.

- If the address > 64K bit in the LSR for a sensed I/O device is off when the program senses the LSR, the system sets the program mode register I/O > 64K bit off.

Because the system uses a two-instruction sequence (LCP, then LIO) to set the address > 64K bit in the LSR for the selected device, and a two-instruction sequence (SNS, then SCP) to inspect the address > 64K bit in the LSR for the selected I/O device, an interrupt may occur between the two instructions in either operation. However, it is not necessary to mask interrupts during these operations as each interrupt level has its own program mode register and, therefore, cannot destroy the contents of another interrupt level PMR.

Byte 1 Bit 6, Storage Protect (Model 15 only)

When this bit is on, the CPU inspects the storage protect keys for each memory cycle except I/O cycles. Violations (unauthorized attempts to use protected areas) cause program check interrupts (or processor checks if interrupt level 7 is not enabled).

When this bit is off, the CPU ignores the protect keys and permits access to all storage locations. This bit must not be set until the address translate table has been properly loaded. Furthermore, this bit must be off whenever the address translate table contents are changed.

Byte 1 Bit 7, Mask Interrupts

When bit 7 is on, all interrupt requests (except program check) remain pending and the CPU remains in its present level. The program can set this in the program level or any interrupt level. The CPU cannot change levels until the program sets the bit off. The program must set bit 7 off before it resets any interrupts. Otherwise, the CPU remains in the current level until the mask is set off. Use this bit with care to avoid overrun in those devices whose interrupts must be serviced within a certain period of time.

Byte 2 Bits 0-6, Reserved

Byte 2 Bit 7, I/O Greater Than 256K

This bit represents the 256K-bit (the bit representing a 262,144 decimal value) in a binary address. The CPU uses this bit while executing LIO and SNS instructions.

- If the I/O > 256K bit is on when the program issues a load I/O instruction to an associated I/O LSR, the I/O > 256K bit in the selected LSR turns on. If the I/O > 256K bit is off when the program issues a load I/O instruction to the LSR, the I/O > 256K bit in the selected LSR turns off.

- If the I/O > 256K bit in the LSR for a sensed I/O device is on when the program senses the LSR, the system sets the program mode register I/O > 256K bit on. If the I/O > 256K bit in the LSR for the selected I/O device is off when the program senses the LSR, the system sets the program mode register I/O > 256K bit off.

Because the system uses a two-instruction sequence (LCP, then LIO) to set the I/O > 256K bit in the LSR for the selected device, and a two-instruction sequence (SNS, then SCP) to inspect the I/O > 256K bit in the LSR for the selected I/O device, an interrupt can occur between the two instructions in either operation. However, it is not necessary to mask interrupts during these operations, as each interrupt level has its own program mode register and, therefore, cannot destroy the contents of another interrupt level PMR.

General Notes About the PMRs

The contents of byte 1 bits 0 through 6 and byte 2 bit 7 are unique for each operating level. Byte 1 bit 7 (mask interrupts) is common to all levels because once byte 1 bit 7 is on, the machine cannot pass to another level; consequently, independent control of the mask bit is not required.

Whenever the program changes the contents of its program mode register, bit 7 prevents or allows interrupts that may occur before the CPU accesses the next sequential instruction. All other bits in the PMR become effective when the CPU accesses the next sequential instruction.

When an operator turns the power switch on, the contents of the PMRs are unpredictable. Therefore, the Model 15 CPU enters privileged mode and disables all other PMR functions. Since the Model 12 CPU has no privileged mode, it disables all the PMR functions at this time. The CPU remains in this state until it accesses the first instruction to load any PMR.

Before any interrupts occur, the program must load the program level PMR and then the interrupt level PMRs. This prevents checks that could otherwise occur from using the indeterminate contents in the PMRs after power up. The program must not set the translate bits (PMR bits 1, 2, and 3) or the storage protect bit (bit 6) until it has loaded the address translate table (ATT) and storage protect registers. Otherwise, use of the unpredictable bits stored in the ATT or storage protect registers after powering up can result in checks. (See *Storage Protect Registers.*) The PMRs must be initialized using an LCP instruction. The CCP instruction does not control the I/O > 256K PMR bit.

Program Status Register (PSR)—All Models

Each model of the system has one program status register (PSR) that operates with the base system. Systems equipped with the Dual Program Feature (never Model 15) are equipped with a second PSR that is associated with program level 2. The leftmost byte of each PSR serves as a length count recall register during interrupts. The rightmost byte of the PSR serves as a condition register during interrupts.

During load register operations that specify the program status register, the CPU sets both the PSR and the condition registers using data from the rightmost byte of the operand. The contents of the PSR remain unaltered until the next branch, insert-and-test-characters, or decimal-type instruction has been executed.

Q-Register—All Models

This register accepts the Q-byte from the instruction. The Q-byte controls operations performed by instructions.

Storage Address Register—All Models

The storage address register (SAR) holds the 2-byte logical address that is to be accessed in main storage.

Storage Data Bus In Register (SDBI)—Models 12C and 15

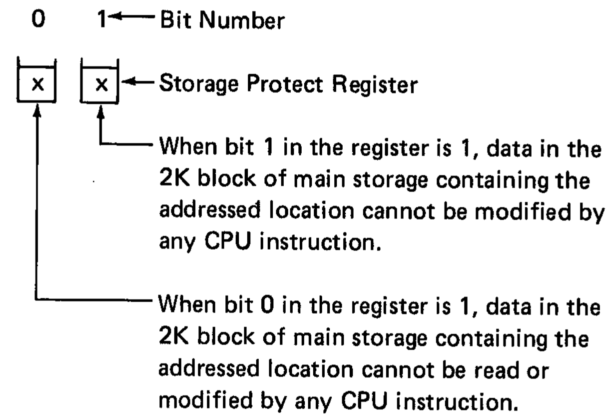
The storage data bus in register serves as a place to store data temporarily as it passes from the arithmetic and logical unit (ALU) to main storage. The program cannot access this register and its contents are not displayed on the display panel.

Storage Data Register (SDR)—Models 8, 10, and 12

This register temporarily stores data moving between the processing portions of the CPU and main storage. Data can enter the storage data register from ALU or from main storage. Data can be sent from the storage data register to either the B-register or to main storage. This register cannot be accessed by the program and its contents cannot be displayed on the display panel.

Storage Protect Registers—Models 15C and 15D Only

Each Model 15C and 15D has one storage protect register for each 2K bytes of main storage. The program loads the storage protect registers by means of the load CPU instruction, and stores their contents in main storage by means of the store CPU instruction. (Storage protect register bits 0 and 1 correspond to bits 0 and 1 in the bytes loaded and stored.) During each machine cycle that is not an I/O cycle the CPU examines the program mode register storage protect bit (bit 6). If the bit is a 1, the storage protect function is active and the 2K block of storage that holds the addressed storage position is subject to control by keys stored in the register. These bits and their meanings are:



The storage protect registers must be initialized after a power down condition before either a storage protect function or an address translation function occurs.

PHASES

The processing unit performs each of its operations in two phases: *instruction phase* and *execution phase.* During the instruction phase, the CPU moves an instruction from storage to the designated CPU registers as follows:

- The op code byte into the op code register.
- The Q-byte into the Q-register.
- The operand 1 address into the BAR, and the operand 2 address into the AAR. If an operand is being addressed by base displacement addressing, the CPU adds the value in the 1-byte operand address portion of the instruction to the address previously loaded into the selected index register before placing the resulting address in the appropriate operand address register (AAR or BAR).

During the execution phase, the CPU executes the instruction just fetched from storage as follows:

- Fetches a byte of data from each operand to be used.
- Examines, moves, or modifies the data as directed by the instruction op code.
- Repeats this process until the operation is complete.

Some instructions combine the phases so that there is no distinct execution phase. These are:

- Branch on condition
- Jump on condition
- Load address
- Advance program level
- Halt program level
- Command CPU
- SIO

CYCLES

Data is moved into and from storage during time intervals called cycles. During single-byte mode operations, the CPU moves 1 byte per cycle; during dual-byte mode operations (for example, during Model 12 and Model 15 high-speed disk read and write I/O operations) the CPU moves 2 bytes of data concurrently during the cycle.

The processing unit uses at least three cycles for each instruction (the 3 bytes in the shortest instruction times one cycle per byte). I/O adapters (sometimes called *attachment features*) also use processing unit cycles when they initiate the transfer of data between the I/O device and main storage. These are called cycle-steal or I/O cycles because the processing unit lets the adapter interrupt regular processing to steal time for I/O cycles between any two processing unit cycles. During this I/O cycle, the processing unit moves 1 byte of data (single-cycle mode) or 2 bytes of data (dual byte mode).

The following list defines the System/3 cycles and the operation performed during each cycle. (Note that the cycle name indicates the phase in which the cycle occurs and the type of operation performed.)

Cycle	Operation
I-Op	The CPU moves the op code from main storage into the op code register.
I-Q	The CPU moves the Q-byte from main storage into the Q-byte register.
I-R	The CPU moves the third byte of a command-type instruction (the R-byte) to attachment or CPU control logic, then the CPU (sometimes aided by the attachment) executes the instruction. *Note:* The R-byte is sometimes called the control byte because it contains information, not available in the Q-byte, that is used by the CPU or attachment during command execution. The CPU does not use I-R cycles for instructions that address storage.

Cycle	Operation
I-X1	The CPU adds the single-byte displacement from the instruction to the contents of the index register specified by the op code and stores the results (the operand 1 address) in the BAR. This cycle is used only when the op code specifies base displacement addressing for operand 1.
I-H1	The CPU moves the high-order (leftmost) byte of the first operand address from main storage into the leftmost half of the BAR. This cycle is used only when the op code specifies direct addressing for operand 1.
I-L1	The CPU moves the low-order (rightmost) byte of the first operand address from main storage into the rightmost half of the BAR. This cycle is used only when the op code specifies direct addressing for operand 1.
I-X2	The CPU adds the single-byte displacement from the instruction to the contents of the index register specified by the op code, then stores the resulting operand 2 address in the AAR. This cycle is used only when the op code specifies base-displacement addressing for operand 2.
I-H2	The CPU moves the high-order (leftmost) byte of the second operand address portion of the stored instruction to the leftmost half of the AAR. This cycle is used only when the op code specifies direct addressing for operand 2.
I-L2	The CPU moves the low-order (rightmost) byte of the second operand portion of the instruction from main storage to the rightmost half of the AAR. This cycle is used only when the op code specifies direct addressing for operand 2.
E-A	The CPU moves a byte of the second operand from storage to the data recall register. These cycles are not used unless two operands are involved in the instruction.
E-B	The CPU moves a byte of the first operand from main storage, operates on it, and returns it to main storage.
I/O	When the CPU is operating in single-byte mode, the CPU moves a single byte of data between main storage and an input/output unit. When the CPU is operating in dual-byte mode, the CPU moves 2 bytes of data, at the same time, between main storage and an input/output unit.

BRANCHING—ALL MODELS

During the instruction phase of each branch instruction, the CPU fetches the branch-to address from the operand address portion of the instruction and loads it into the address recall register. At the end of the instruction phase of the branch instruction, the address of the next sequential instruction resides in the instruction address register. During execution of this instruction, the CPU:

- Swaps the contents of the address recall register with the contents of the contents of the instruction address register if a branch is to occur.
- Retains the contents of the address recall register and the instruction address register as they were loaded during the instruction phase if a branch is not to occur.

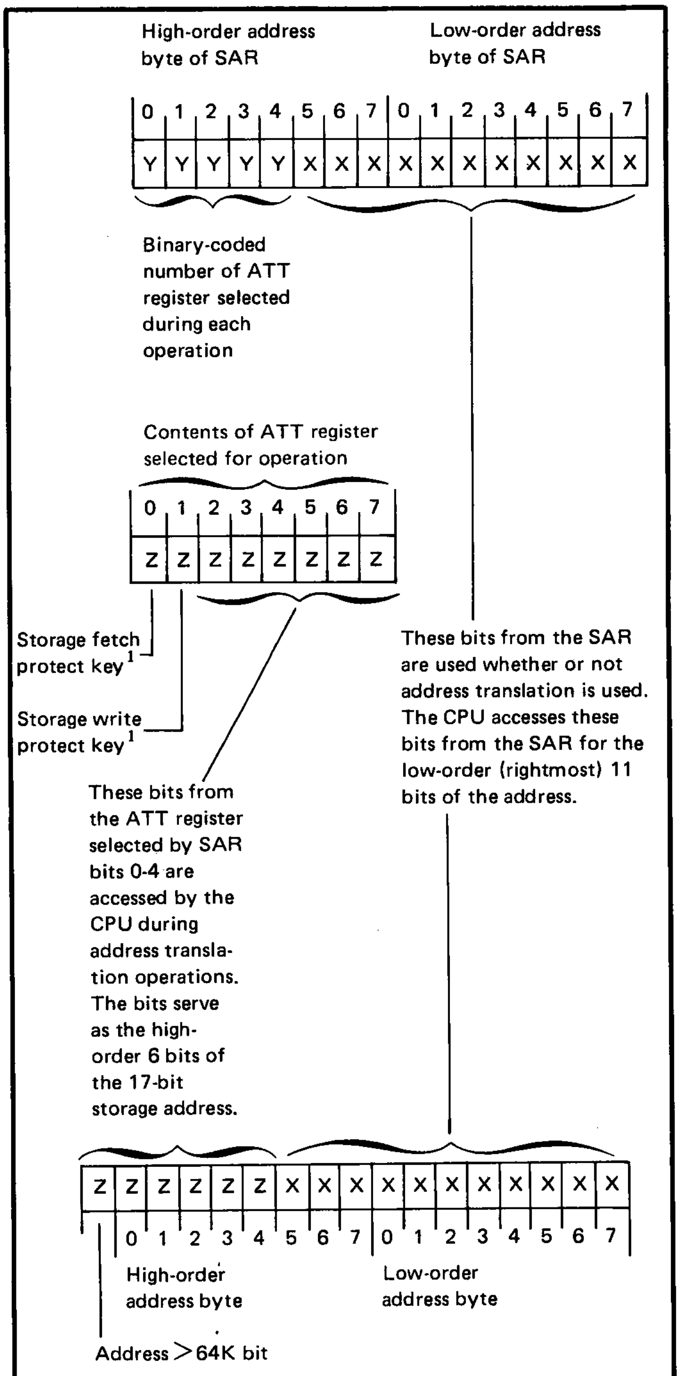
ADDRESS TRANSLATION—MODELS 12C AND 15

System/3 uses a 2-byte (16-bit) address and a 1-byte wide data path. Using a 16-bit address limits addressable storage to 64K (65,536) positions. In order to address additional main storage positions, an address translation table (consisting of 32 8-bit registers is required, which resides between the 16-bit storage address register (SAR) and main storage. As shown in Figures 3-5 and 3-6 the address translation table converts 16-bit logical addresses from the storage address register into 17, 18, and 19 bit real addresses (physical addresses).

The addresses in SAR are considered to be logical addresses because they run from 0 to 64K, and are not uniquely associated with the actual storage location accessed. The addresses applied to storage are considered to be real or physical addresses.

INPUT/OUTPUT CHANNEL, I/O ATTACHMENTS, AND I/O CHANNEL ORGANIZATION

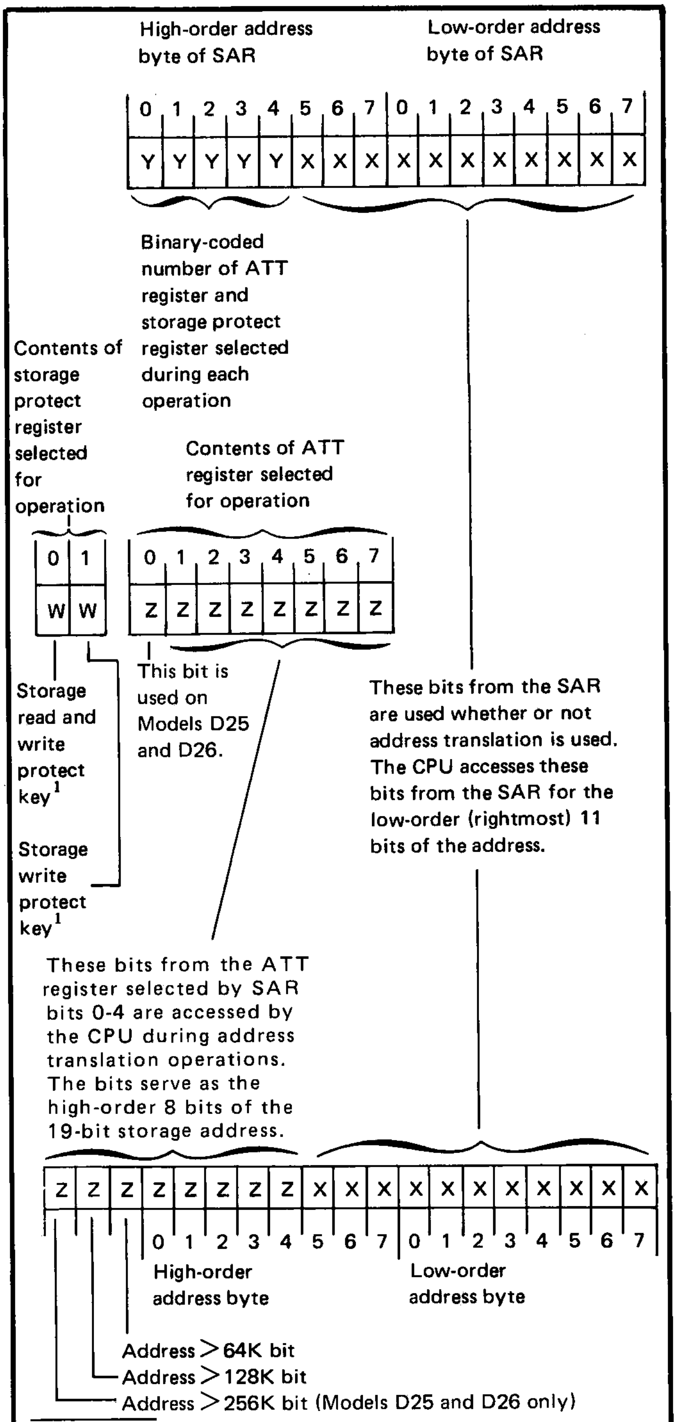
Each I/O unit has an attachment, sometimes called an adapter. The processing unit has a single I/O channel to which all the attachments connect. The channel serves as a data and instruction path between the processing unit and the attachment circuits of all the I/O units. The processing unit is designed to communicate with only one I/O device at a time; therefore, each I/O device has an assigned priority for channel use.



[1]Storage is protected if bit = 1 (not used on the Model 12).

Note: One ATT register is always selected for each operation. If address translation is not specified, the CPU uses the entire contents of the SAR as the physical address of storage. If address translation is specified, the CPU uses address bits from the SAR and the specified ATT register, as shown in this figure. In each case, the CPU examines the fetch and write protect keys in the ATT.

Figure 3-5. How Address Translate Table is Used for Address Translation on the 5412 Model C and 5415 Models A and B, and Storage Protection Functions on 5415 Models A and B



[1]Storage is protected if bit = 1.

Note: One ATT register and one storage protect register are always selected for each operation. If address translation is not specified, the CPU uses the entire contents of the SAR as the physical address of storage. If address translation is specified, the CPU uses address bits from the SAR and the specified ATT register, as shown in this figure. In each case, the CPU examines the storage protect keys in the storage protect register.

Figure 3-6. How Address Translate Table Is Used for Address Translation and Storage Protection Functions on 5415 Models C and D

CYCLE STEAL OPERATIONS

System/3 overlaps processing operations with I/O operations so that processing can continue while I/O operations occur. To do this, the processing unit lets the I/O attachment interrupt regular processing to use a machine cycle as an I/O cycle, then the processing unit returns to regular processing operations by handling the next processing unit cycle as if processing had not been interrupted. Because the attachment used a machine cycle that would otherwise be used for normal processing unit operations, the attachment is said to have *stolen* the cycle from the processing unit operation. I/O cycles occur whenever a byte of data (2 bytes, in dual byte mode I/O operations) must be moved between main storage and the I/O unit. The operation during which I/O cycles occur is called a *cycle steal* operation.

I/O DEVICE CONTROL

The following instructions control I/O devices:

Start I/O
Load I/O
Sense I/O
Test I/O and Branch
Advance Program Level

Each instruction specifies the operation being performed, the device or unit performing the operation, and the addressing scheme used for the instruction. Because the exact operations performed are different for each attached device, this manual describes, in detail, the five I/O instructions associated with each separate I/O device in the chapter about that device.

INTERRUPTS

The processing unit performs operations by executing sequential instructions until the instruction sequence is altered by a programmed branch or interrupt. To avoid contention, the CPU processes interrupts according to a predefined priority (see Figure 3-7).

All interrupts follow the same general outline; (1) interrupt the program in progress (always when a device becomes ready (Model 15) or at op-end), (2) execute the requested program, (3) return control to the interrupted program.

Each interrupt level has a separate IAR, ARR and (Models 12C and 15) PMR in the CPU so these registers for the main program are not disturbed. The condition register and any index register used during the interrupt must be stored at the beginning of the interrupt routine and reestablished at the end of the same routine.

The interrupt routine being performed is established by the interrupt priority latches. As in cycle steal, the highest interrupt level device takes precedence over lower level devices. Thus, it is possible for an interrupt routine to interrupt a routine of a lower priority device. However, each device maintains its interrupt request until it is satisfied, so the lower priority device finishes its routine upon completion of the higher level routine.

The stored program controls the ability of a device to interrupt by enabling and disabling the device through SIO, LIO, or CCP instructions. Once an interrupt has occurred, the interrupt routine is also ended by the same type of instruction. (Models 8, 10, and 12 interrupts cannot be enabled or disabled by LIO instructions.)

Figures 3-8 and 3-9 show programming for typical interrupt routine.

Supervisor Program (Interrupt Level 0)—Model 15 Only

A user program passes processing unit control to the supervisor by initiating an interrupt on level 0. To do this, use the command CPU instruction with a Q-code of 10 (supervisor call).

Op End Interrupt (Interrupt Level 5)—Model 15 Only

When an I/O device has completed the operation in progress, its attachment sends an op-end request to the processing unit if op-end interrupt is enabled for the device.

At the end of the execution phase for the instruction being executed by the processing unit when any attachment requests an op-end interrupt, the CPU switches to the interrupt level 5 IAR for the address of the next sequential instruction.

Once processing unit control is transferred to the interrupt level 5 program, that program determines which device initiated the op end interrupt. This is done through TIO and SNS instructions. The method for determining which device requested the interrupt and what priority each device holds, is completely a programming function.

<table>
<tr><th>Interrupt Level</th><th>Priority</th><th>Function Performed by Model 15</th><th></th><th>Function Performed by Models 8, 10, and 12</th></tr>
<tr><td>7</td><td>1</td><td colspan="2">Program check—handles soft errors.</td><td>None</td></tr>
<tr><td>6</td><td>2</td><td>Interval timer and not-ready-to-ready interrupt</td><td rowspan="4">I/O control and data transfer</td><td rowspan="4">Same as Model 15, except interrupt level 6 is not used</td></tr>
<tr><td>4</td><td>3</td><td>SIOC</td></tr>
<tr><td>3</td><td>4</td><td>BSCC/MLTA</td></tr>
<tr><td>2</td><td>5</td><td>Adapters for synchronous communications devices like BSCA, ICA</td></tr>
<tr><td>5</td><td>6</td><td colspan="2">Device op-end—notifies the CPU that the I/O device has reached end of operation or 3741 not ready-to-ready interrupt.</td><td>None</td></tr>
<tr><td>1</td><td>7</td><td colspan="2">Display screen and keyboard—I/O control and data transfer</td><td>Data entry keyboard or printer keyboard—I/O control and data transfer</td></tr>
<tr><td>0</td><td>8</td><td colspan="2">Supervisor program—transfers control from a problem program to the supervisor program.</td><td>Dual programming control—interrupt key initiated functions</td></tr>
<tr><td>None</td><td></td><td colspan="2">Main program level</td><td>Program levels 1 and 2</td></tr>
</table>

Figure 3-7. Interrupt Levels and Priorities

Program Check Interrupt (Interrupt Level 7)—Model 15 Only

To allow efficient multiprogramming, errors caused by one user must not stop the system and deprive all users of processing time. In the Model 15, program check interrupt allows the processing unit to enter an interrupt routine rather than a processor check hard stop condition. Errors causing a program check interrupt are:

- Invalid address
- Invalid Q
- Invalid op
- Privileged op
- Storage violation

The occurrence of any of these errors while in the program check interrupt routine causes a processor check hard stop. Invalid address during I/O cycles also causes a processor check hard stop.

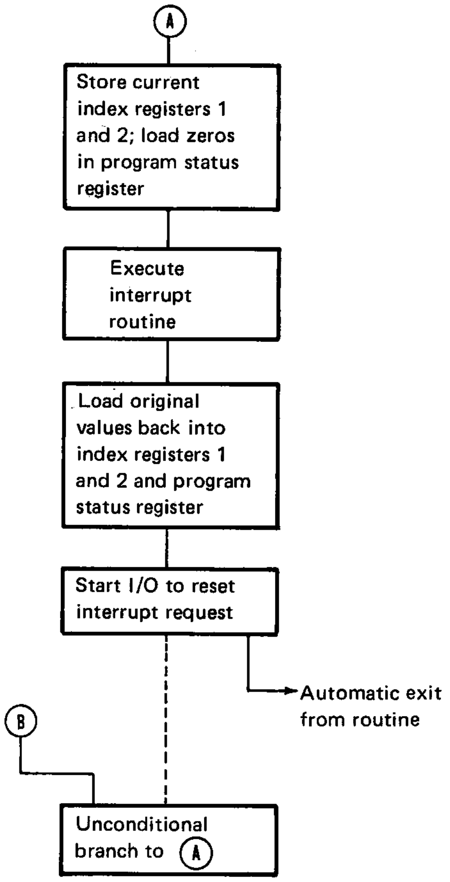
The program check interrupt routine can analyze the cause of the error from status provided by the processing unit hardware, prepare a message for the user causing the error, and then transfer processing unit control to another user thereby making maximum use of the processing unit time.

The program check interrupt is assigned to level 7 which is the highest priority interrupt. The program check function must be enabled by a command processing unit instruction. If the function is not enabled, the checks mentioned cause a processor check hard stop. The command processing unit instruction is also used to disable and reset interrupt level 7. The status required to analyze the error source is provided in registers that may be stored using the store processing unit instruction. This status includes the specified check, the physical main storage address at the time of error, and the active interrupt level, if any, at the time of error.

INTERRUPT MASK—MODELS 12C AND 15 ONLY

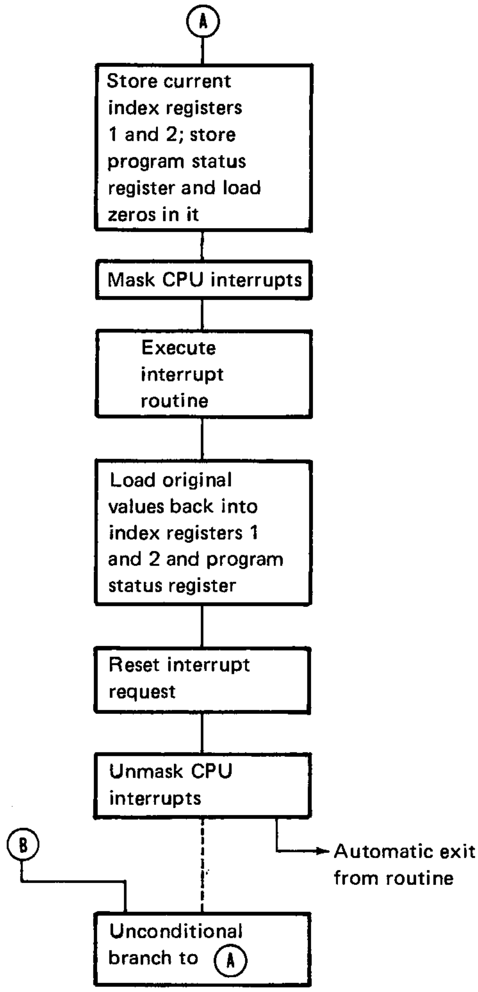
A mask function is provided to simplify interrupt processing. This function gives the programmer the ability to complete a critical routine before it is interrupted by a higher priority program. The mask interrupt function is controlled by a bit in the PMR. When this bit is on, any higher priority interrupt request remains pending until the mask is set off. The exception is interrupt level 7 for the Model 15 (program check interrupt). It is not affected by the mask.

The interrupt mask bit in the PMR must be set off before an interrupt level is reset. Failure to do so will cause the processing unit to remain in that interrupt level.



Note: The interrupt instruction address register must be set to the address of (A) or (B) before the first interrupt occurs. The normal operation of the processing unit will leave the interrupt instruction address register at the address of (B) at the end of the interrupt routine.

Figure 3-8. Typical Interrupt Routine for Models 8, 10, and 12B Programs



Note: The interrupt instruction address register must be set to the address of (A) or (B) before the first interrupt occurs. The normal operation of the processing unit will leave the interrupt instruction address register at the address of (B) at the end of the interrupt routine.

Figure 3-9. Typical Interrupt Routine for Models 12C and 15 Programs

Chapter 4. System Control Panel

The system control panel (Figure 4-1) contains the switches and lights required to operate and control the system. System controls are divided into three sections: operator controls, customer engineering (CE) controls, and console display.

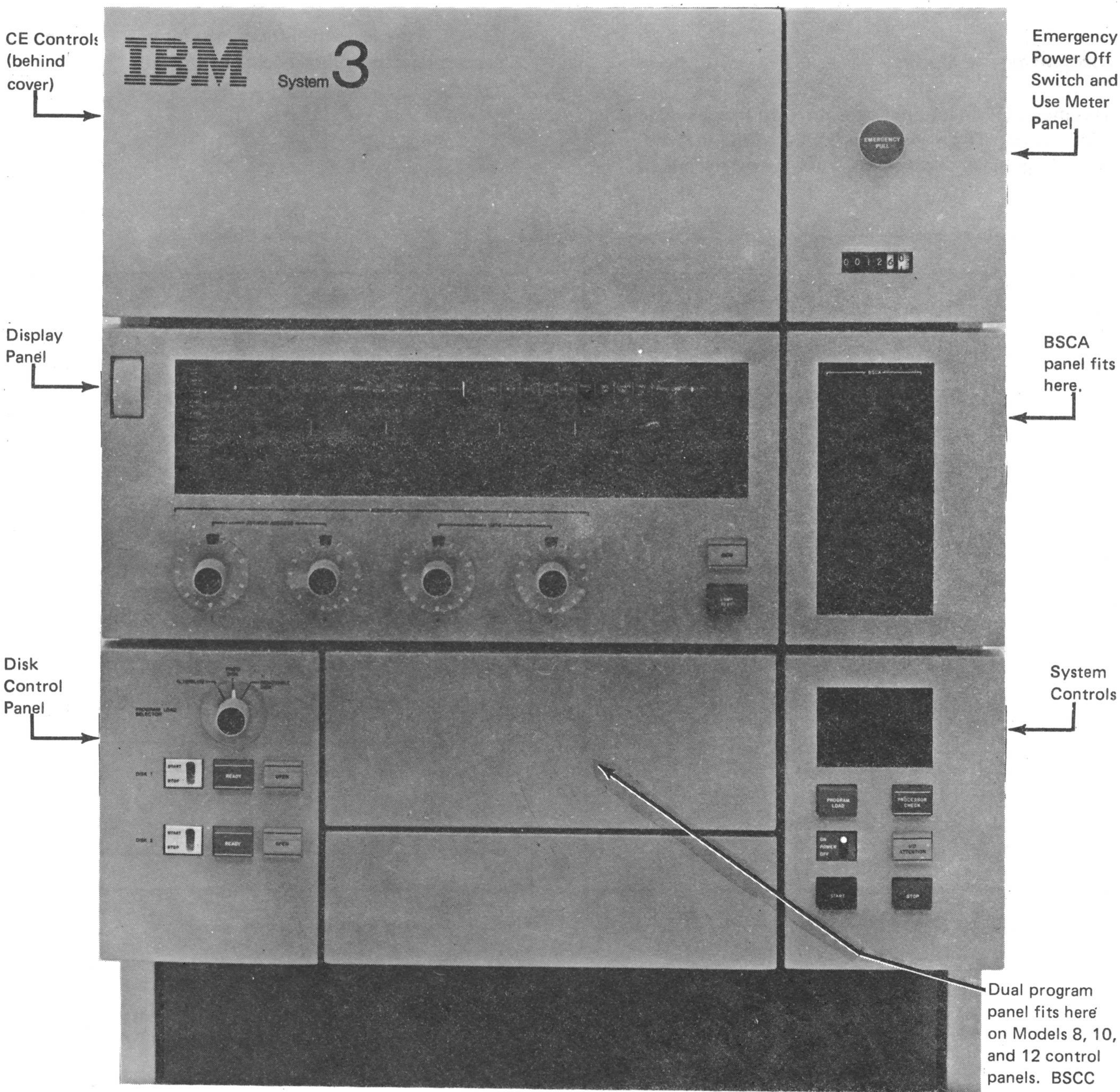


Figure 4-1. Example of System Control Panel (Model 15A shown)

OPERATOR CONTROLS

Emergency Power Off and Meter Panel

Emergency Power Off—All Models

Pulling this switch (Figure 4-2) in an emergency removes power from the processing unit and most I/O units. Once pulled, the switch remains locked in the off position. Power can be restored to the system only by intervention of maintenance personnel. Data in storage is lost any time power is dropped.

Usage Meter—All Models

This meter (Figure 4-2) records the time that the system is in operation. The meter records all the time that the processing unit is in operation from the time someone presses the START key or LOAD key until the job is complete. Whenever the system is performing I/O operations during a programmed halt, the meter records time until all I/O operations end. Time is not recorded while:

- The processing unit is in a halt state because of either a manual halt or a programmed halt.
- The processing unit is not operating because it has stopped with a processor check.
- System power is off for any reason.
- The CE is servicing the processing unit or is using the processing unit while servicing an I/O unit.

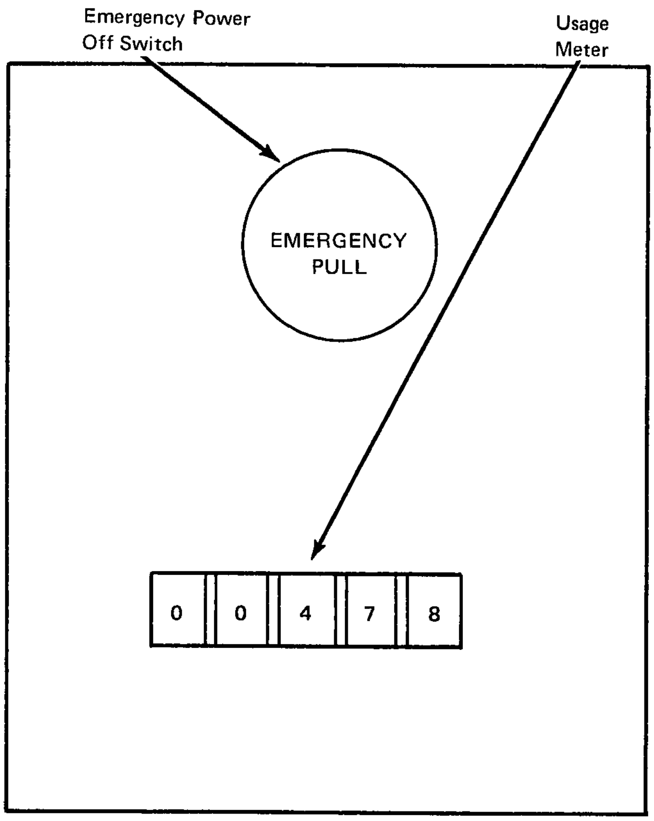


Figure 4-2. EPO and Meter Panel

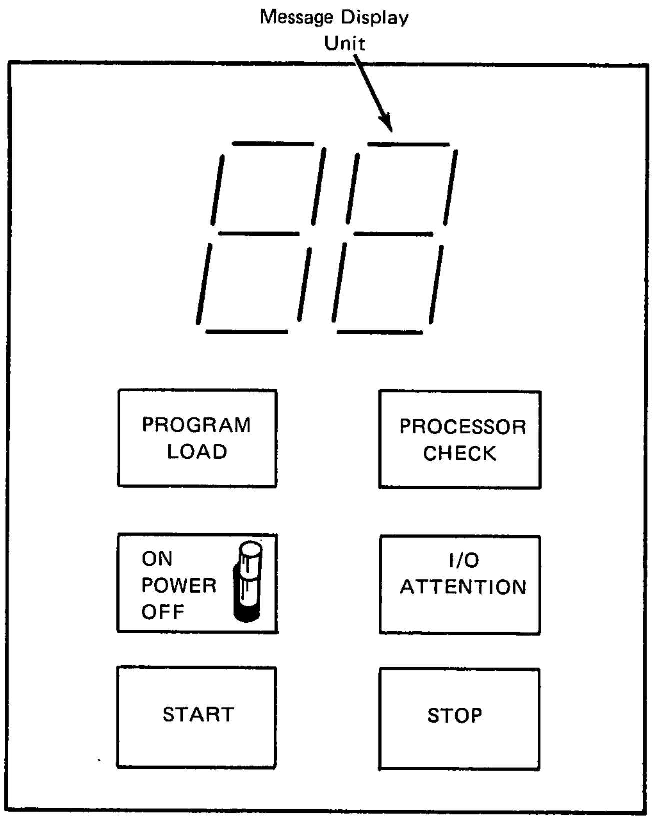


Figure 4-3. System Controls

SYSTEM CONTROLS

PROCESSOR CHECK Light—All Models

All processor checks turn the PROCESSOR CHECK light (Figure 4-3) on immediately and cause a processor check stop.

Initiating a system reset, operating the CHECK RESET key on the CE panel, or performing an IPL operation turns this light off. The checks that light this indicator cause the processing unit to stop immediately. When the processing unit stops, data from any I/O operation that is in progress is lost. The specific check that caused the processing unit to stop is indicated in the display panel section of the system control panel. Processor checks that can occur are listed in Figure 4-9.

Message Display Unit—All Models

This two-position display unit (Figure 4-3) keeps a running display of the halt identifier portion of a halt-program-level instruction. The left half of the unit displays a character specified by the second byte of the instruction; the right half displays a character specified by the third byte of the instruction.

If the system is equipped with the Dual Program Feature, there are two message display units on the control panel, one for each program level.

I/O ATTENTION Light—All Models

I/O ATTENTION light (Figure 4-3) turns on when an addressed I/O unit requires operator attention. Processing unit operation does not stop, but the I/O unit requiring attention will not accept a start I/O instruction until the condition is corrected. The I/O unit requiring attention usually turns on an indicator to show what attention is required.

On Models 8, 10, and 12, a disk drive not ready condition also turns on the I/O ATTENTION light and results in the following action:

- If the system is using the Dual Program Feature, a program level advance occurs.
- If the system is not using the Dual Program Feature, the program loops on the instruction until the drive becomes ready, then executes the instruction.

After the operator has performed the required service on the I/O unit needing attention, the I/O ATTENTION light turns off.

The following conditions are typical of conditions that turn on the I/O ATTENTION light on any model:

- Forms run-out
- Hopper empty
- Stacker full
- Chip box full
- Cover open

The following 3340 conditions also turn on the Model 12 I/O ATTENTION light:

- Drive not ready (indicated by no 3340 READY light)
- Wrong data module size (no indicator)
- Attempt to write on module set up for read-only mode (indicated by read-only indicator)

POWER ON/OFF Switch—All Models

This toggle switch (Figure 4-3) controls the power to the system unless the emergency power off switch has been pulled. Turning the POWER switch on initiates power on system reset.

Good practice dictates pressing the STOP switch *before* turning the POWER switch off; otherwise, stored data cannot be considered to be valid should the POWER switch be turned off.

PROGRAM LOAD Key—All Models

Pressing PROGRAM LOAD causes the processing unit to perform the following specific actions:

- Perform a system reset operation.
- On all models except Model 15, reset the program level 1 instruction address register to 0. On Model 15 only, reset the program level instruction address register to 0.
- On all models except Model 15, reset the program level 1 program status register (and, if the system is equipped with the Dual Program Feature, the program level 2 program status register) to 0. On Model 15 only, reset the program status register to 0.
- Reset the data address register for the device selected by the program load selector switch to 0.

When the PROGRAM LOAD key is released, the processing unit executes the instruction read into storage starting at location 0000. (*Exception:* For performing an IPL from the 3340, see *PROGRAM LOAD SELECTOR Switch*.)

If the selected I/O device is not ready, the I/O ATTENTION light turns on when the PROGRAM LOAD key is pressed. It is necessary to make the I/O device ready to complete the program load function.

For additional information on program loading from the 5444, 1442, 5424, 2560, 3741, or 3340, refer to appropriate I/O section of this manual.

Models 12 and 15 Program Note

If the CE MODE SELECTOR switch is set at PROCESS MODE when power is supplied to the system, the first time either PROGRAM LOAD or SYSTEM RESET is used, the two characters set into the console data switches are read into every position of storage.

STOP Key/Light—All Models

Pressing STOP stops the processing unit at the end of the operation being performed and turns on the STOP light. The processing unit completes all I/O data transfer in process when the key was pressed without loss of data.

To restart processing, press the START key.

START Key—All Models

This key turns off the STOP light and starts the processor. The processing unit resumes execution of the program being executed when the STOP key was pressed; execution resumes at the next sequential instruction.

The START key is also used during diagnostic operations when the processing unit is operating in CE mode. In this mode, the processing unit performs (1) a complete instruction cycle, (2) a machine cycle, or (3) a clock cycle each time the START key is pressed. If the START key is pressed when the system is not executing a machine cycle, an I-op cycle for the instruction addressed by the current IAR is executed.

DISK CONTROLS

PROGRAM LOAD SELECTOR Switch—Models Using 5444

This switch (Figure 4-4) selects the source from which the program is to be loaded. One of three sources can be selected:

- *Removable Disk:* 5444 drive 1, removable disk, track sector address 00000
- *Fixed Disk:* 5444 drive 1, fixed disk, track sector address 00000
- *Alternate:* First record of alternate device

A program load operation from disk initiates the loading of the first 256-byte sector from disk location 00000 into main storage, starting at address 00000. The sector identifier field is not compared and the DDCR is not changed.

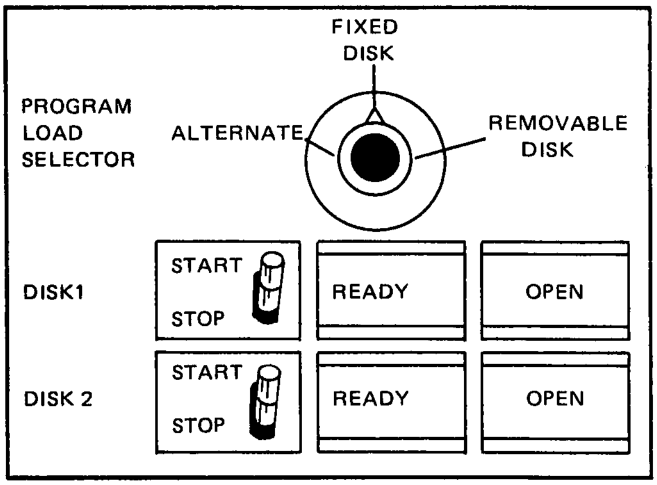


Figure 4-4. Disk Control Panel—Systems Using 5444

OPEN Lights—Models Using 5444

Each 5444 drive has an OPEN light (Figure 4-4) that indicates when the associated drive drawer can be opened for changing the removable disk. This light turns on when the START/STOP switch is turned to the stop position, the read/write head has been retracted, and the disk has come to a stop.

READY Lights—Models Using 5444

Each 5444 drive has a READY light (Figure 4-4) that turns on when the associated drive is ready for use. If operation of the drive is attempted before this light turns on, the I/O ATTENTION light on the control panel turns on.

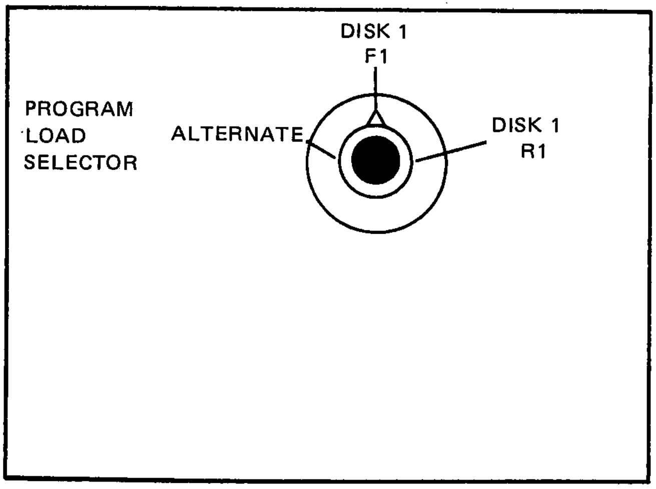
PROGRAM LOAD SELECTOR Switch—Models Using 3340/3344

This switch (Figure 4-5) selects the source from which the program is to be loaded:

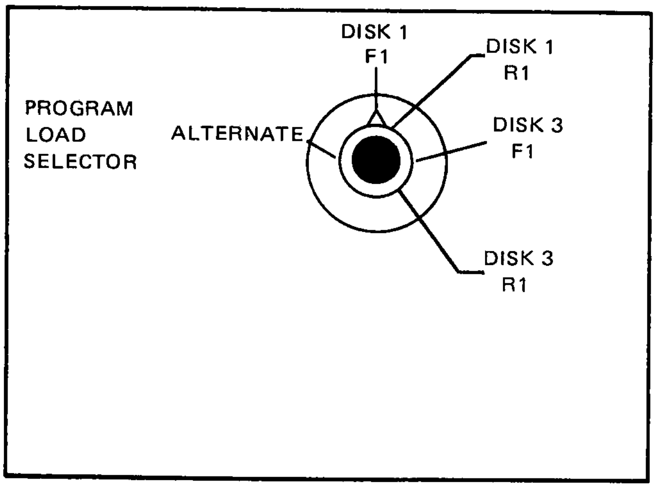
- *Alternate:* First record of alternate device. (This position also sets the 3340/3344 sense byte 1, bit 2 to 0.)
- *Disk 1 F1:* 3340 drive 1, cylinder 0, head 0, record 48. (This position also sets the 3340/3344 sense byte 1, bit 2 to 0.) If IBM programming support is used, the system loads $$SPVR from the first 5444 simulation area on the disk when PROGRAM LOAD is pressed.
- *Disk 1 R1:* 3340 drive 1, cylinder 0, head 0, record 48. (This position also sets the 3340/3344 sense byte 1, bit 2 to 1.) If IBM programming support is used, the system loads $$SPVR from the second 5444 simulation area on the disk when PROGRAM LOAD is pressed.
- *Disk 3 F1:* 3344 drive 3, cylinder 0, head 0, record 46. (This position also sets the 3340/3344 sense byte 1, bit 2 to 0.) If IBM programming support is used, the system loads $$SPVR from the D3A 5444 simulation area on the disk when PROGRAM LOAD is pressed.
- *Disk 3 R1:* 3344 drive 3, cylinder 0, head 0, record 46. (This position also sets the 3340/3344 sense byte 1, bit 2 to 1.) If IBM programming support is used, the system loads $$SPVR from the D3B 5444 simulation area on the disk when PROGRAM LOAD is pressed.

When the 3340/3344 is selected as the primary source, pressing the PROGRAM LOAD key initiates the following series of actions:

1. The system initializes the attachment.
2. The attachment loads control storage from either 3340 drive 1 or 3344 drive 3.
3. The attachment reads record 48 (3340) or record 46 (3344) as the final step in the IMPL (initial micro-program load) process. (This record must contain the program link to the system control program.)



Systems Using 3340



Systems Using 3344

Figure 4-5. Disk Control Panel

CONSOLE DISPLAY PANEL

Console Address and Data Switches

These are four 16-position rotary switches on the display panel (Figures 4-6, 4-7, and 4-8). Each position on a switch represents one of the 16 hex digits. The switches are used in conjunction with switches on the CE panel to manually enter data into storage, to manually set up addresses for accessing storage positions, or to manually set up storage addresses at which the program will stop executing instructions.

Models 8, 10, 12, and 15 use the four rotary switches to alter the contents of the storage address register and to perform address compare operations. In addition to the four rotary switches, the Model 12C uses the > 64K ADDR BIT toggle switch (on the CE panel) and the Models 15C and 15D use both the > 64K ADDR BIT and the > 128K ADDR BIT toggle switches (on the CE panel) to alter the contents of the storage address register and to perform address compare operations.

Note: The Model 15 D25 and Model 15 D26 use the EXTENDED SAR ADDRESS BITS rotary switch located on the CE panel.

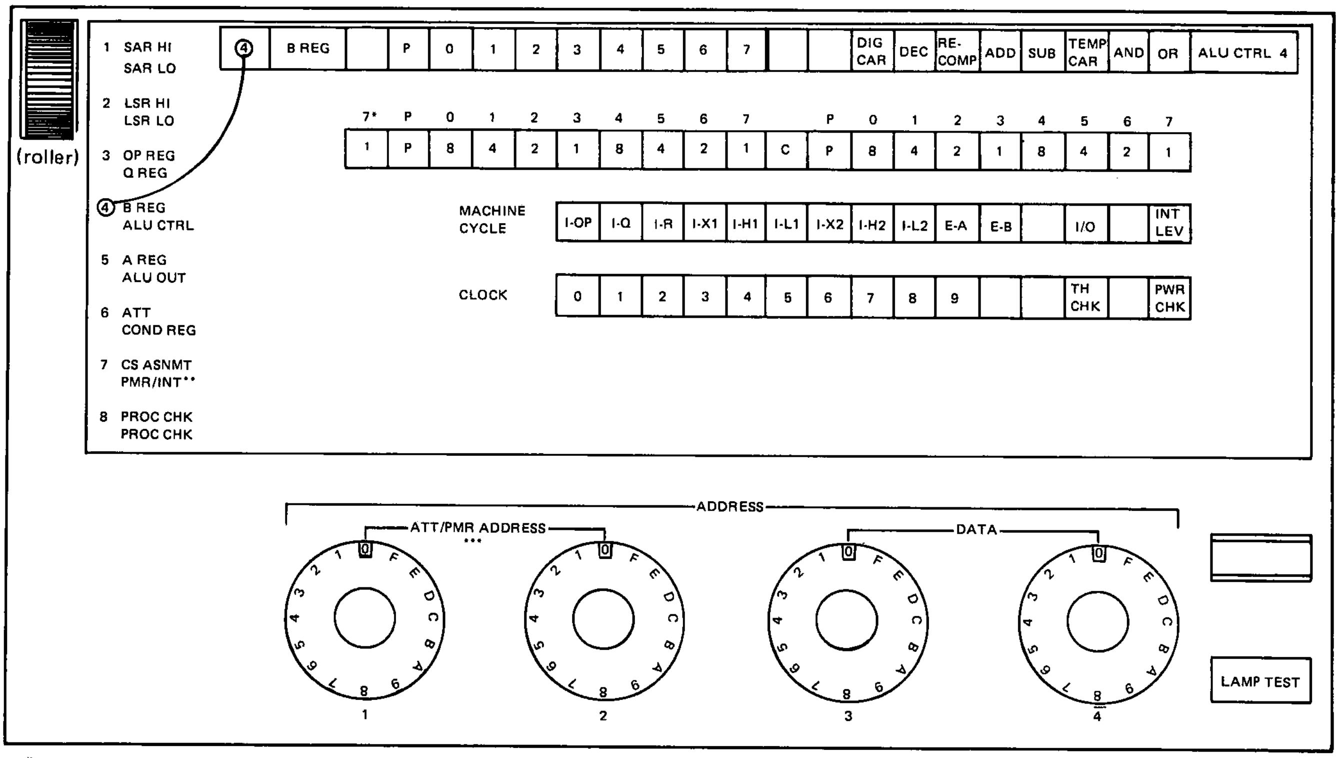
On Models 12C and 15, the two leftmost rotary switches are used to identify the ATT or PMR register whose contents are to be altered or displayed.

The two rightmost rotary switches (identified as DATA switches) are used to enter a single byte of data (2 hex characters) into the specified storage position or register.

Note: Data entered by these switches can be retrieved by the program with a sense I/O instruction that has a Q-byte of hex 00. The data is stored in the 2-byte field specified by the operand 1 address in the instruction.

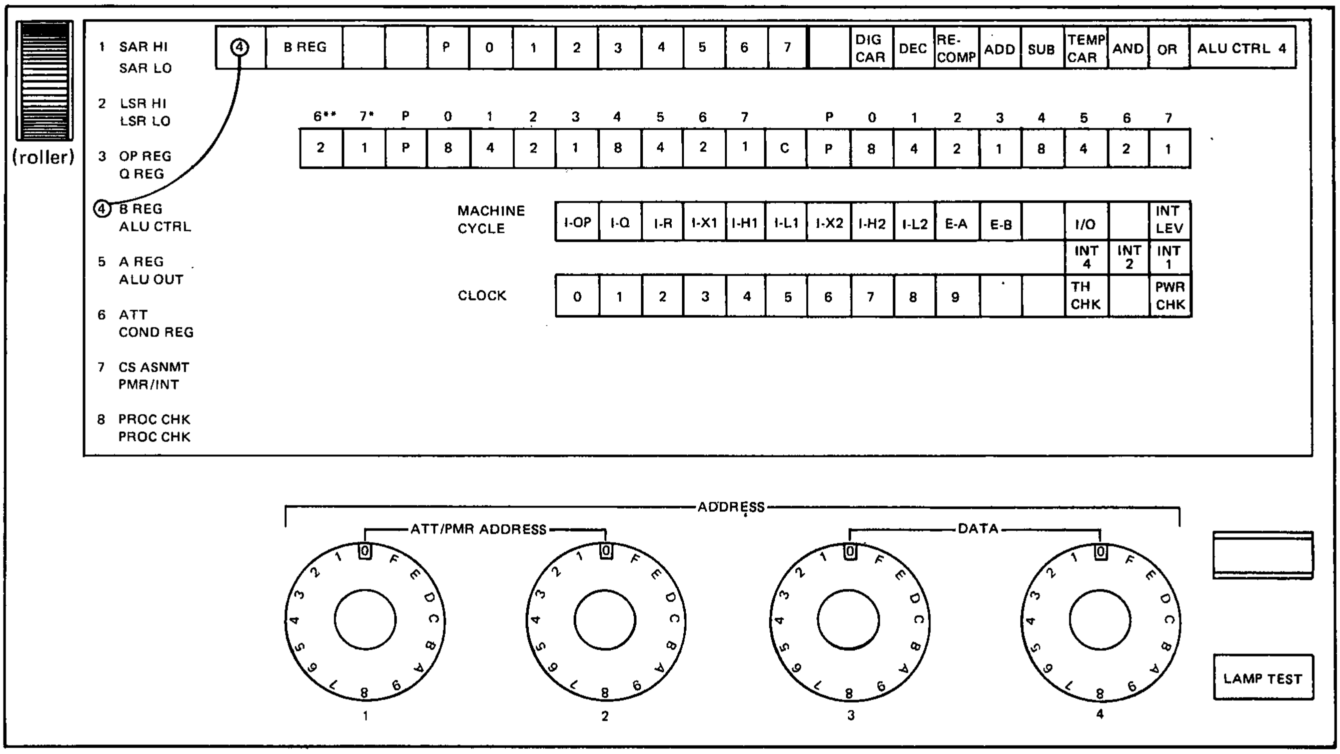
Register Display Unit

The register display unit (Figures 4-6, 4-7, and 4-8) consists of a row of 20 lights and eight legend strips mounted on an 8-position roller-type switch. Turning the roller selects the legend strip and the register to be displayed. The legend strips display the information described by Figure 4-9.



*This bit light, when on, specifies an address greater than 65,536. It appears on Models 12C and 15 panels only.
**INT on Models 8, 10, and 12B.
***Not on Model 12C.

Figure 4-6. Display Panel (All Models Except Models 15C and 15D)



*This bit light, when on, specifies an address greater than 65,536 bytes. Models 8, 10, and 12B panels do not have this indicator.
**This bit light, when on, specifies an address greater than 131,172 bytes.

Figure 4-7. Display Panel (Models 15C and 15D Except Models D25 and D26)

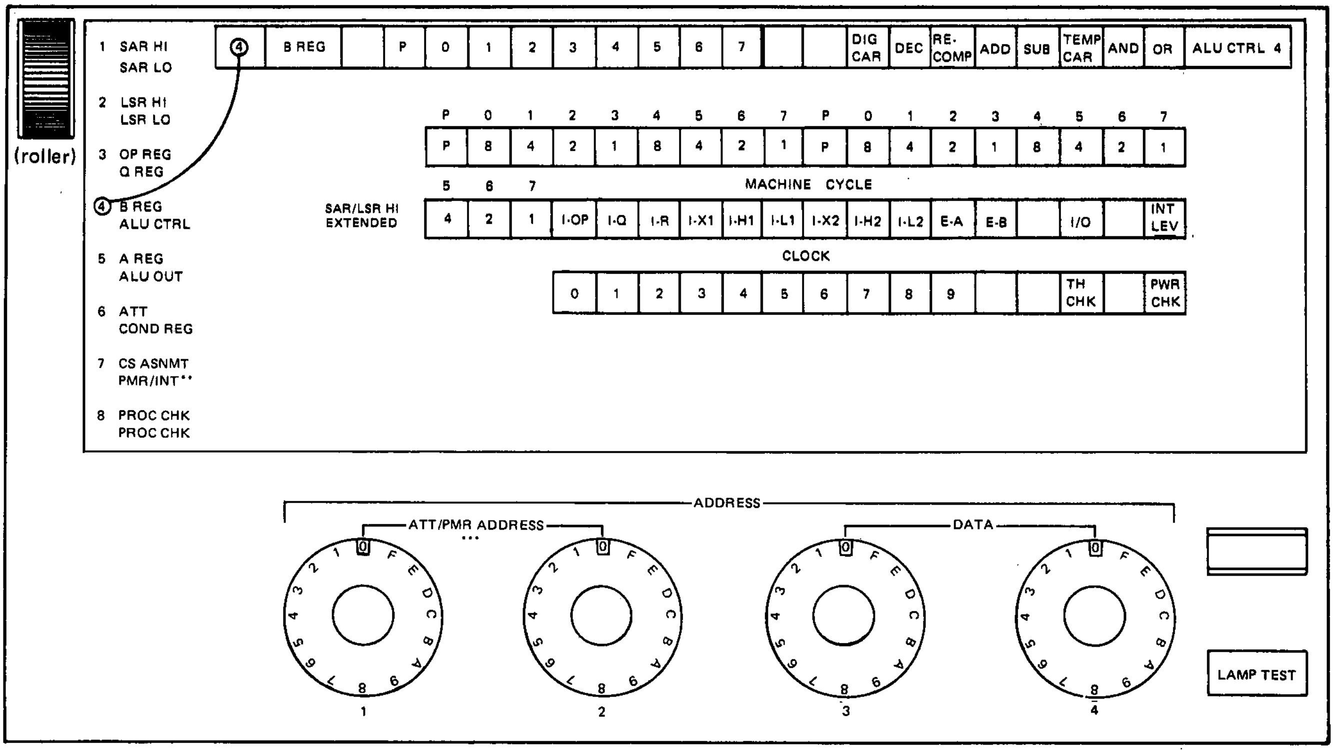


Figure 4-8. Display Panel (Model 15 D25 and Model 15 D26)

Strip Number	System/3 Model	Identification	Information Displayed
1	All	SAR HI/SAR LO	Contents of storage address register (on Model 12C and Model 15, SAR DISPLAY toggle switch must be set at SAR)
2	All	LSR HI/LSR LO	Contents of register selected by setting of LSR DISPLAY SELECTOR switch
3L	All	OP REG	Contents of the op register
3R	All	Q-REG	Contents of the Q-register
4L	All	B-REG	Contents of the B-register
4R	All	ALU CTL	The state of the following ALU controls: DIG CAR (digital carry) DEC (decimal) RE COMP (recomplement) ADD (addition) SUB (subtraction) TEM CAR (temporary carry) AND (logical and) OR (logical or)
5L	All	A-REG	Contents of the A-register
5R	All	ALU OUT	Output of the ALU
6L	8, 10, 12B	Reserved	
	12C, 15	ATT	Contents of ATT (The ATT displayed is the active ATT register unless the alter/display ATT function is being used, in which case the addressed ATT register is displayed. An ATT is always selected and displayed here regardless of whether the contents are being used.)
6R	All	COND REG	The contents of the condition register are displayed as follows: BIN OVF (binary overflow) TF (test false) DEC OVF (decimal overflow) HI (high) LO (low) EQ (equal)
7L	All	CS ASNMT	Cycle steal assignment is displayed as it is presented to the I/O devices on the I/O interface.
7R	8, 10, 12B	INT LEV	Interrupt level, indicating which I/O device is interrupting the program. Level is displayed as a binary encoded value. Interrupt level 0 is indicated as no light in any of the 3 interrupt level code bits and the INTERRUPT CYCLE light on.
	12C	PMR/INT	Program mode register (PMR) and interrupt level. The PMR displayed is the active PMR unless the alter/display PMR function is being used, in which case the addressed PMR is displayed. Interrupt levels are indicated as follows: **Interrupt Level** — **Indicators On** 0 — INT LEV 1 — INT 1 2 — INT 2 3 — INT 3 4 — INT 4

Figure 4-9 (Part 1 of 3). Information Displayed on Legend Strips

Strip Number	System/3 Model	Identification	Information Displayed
	15	PMR/INT (Models A and B) PMR (Models C and D)	Program mode register (PMR) contents and binary encoded interrupt level. The PMR displayed is the active PMR unless the alter/display PMR function is being used, in which case the addressed PMR is displayed. Interrupt level is displayed as a binary encoded value. Interrupt 0 is indicated by no light in all 3 interrupt level code bits and the INT LEV light on. (On Models C and D only, the binary value displayed on the INT 1, INT 2, and INT 4 lights below the MACHINE CYCLES lights serve as the interrupt level code bits.)
8	8, 10, 12B	PROC CHK	The processor checks are displayed as follows: I/O LSR: I/O attachment made an LSR selection error. If LSR F1 or LSR F2 is not on, the LSR is associated with the 1403, 1442, 5203, or 5424. LSR F1: The output from the 3340, 3741 (IPL), or BSCA-1 LSR contained a parity error. LSR F2: The output from an LSR associated with an I/O device is not listed for LSR F1. LSR HI: High-order (leftmost byte) of LSR output has parity error. LSR LO: Low-order (rightmost) byte of LSR output has parity error SAR HI: High-order (leftmost) byte of storage address register has parity error. SAR LO: Low-order (rightmost) byte of storage address register has parity error. INV ADDR: Storage address register contains address that exceeds installed storage capacity. SDR: Storage data register has incorrect parity. CAR: Carry from ALU is wrong. CPU DBO: Processor tried to send data with incorrect parity to an I/O device. OP/Q: Incorrect parity in op-code register or Q-register. INV OP: Invalid op code in op-code register. CHAN DBO: CPU sent data with correct parity to I/O device, but I/O device received data with incorrect parity. INV Q: Invalid Q-byte in the Q-register. DBI: CPU received data containing incorrect parity from an I/O device. A/B: A or B register has incorrect parity. ALU: ALU output has incorrect parity.
	12C, 15	PROC CHK	The processor checks are displayed as follows: I/O LSR: Selection of an LSR by an I/O device was not performed correctly. LSR: Parity is incorrect on the output of the LSR.

Figure 4-9 (Part 2 of 3). Information Displayed on Legend Strips

Strip Number	System/3 Model	Identification	Information Displayed
	12C, 15	PROC CHK (continued)	SAR ATT: Parity is incorrect in the storage address register or in the ATT register located in the processing unit. MSAR: Parity is incorrect at the memory end of the storage address lines. INV ADDR: The MSAR contains an invalid address; that is, the storage address exceeds the system storage size. STOR PROT: An attempt was made to read or write into a protected address (Model 15 only). SDBI: Parity is incorrect at input to storage. SDBO: Uncorrectable data error at output of storage. CAR: Carry out of the ALU is incorrect. DBI: Parity is incorrect on the processing unit end of the data bus in coming from the I/O devices. A/B: Parity is incorrect in the A-register or B-register. ALU: ALU output has incorrect parity (Model 12C only). CPU DBO: Parity is incorrect on the processing unit end of the data bus out going to the I/O devices. OP/Q: Parity is incorrect in the op-register or Q-register. PRIV OP: An attempt was made to execute a privileged operation while in nonpriviledged mode (Model 15 only). INV OP: An invalid op code exists in the op-register. CHAN DBO: Parity is incorrect on the I/O device end of the data bus out coming from the processing unit. INV Q: An invalid Q-byte is present in an I/O instruction. If both this light and the PRIV OP light are on, the check is caused by a privileged op detected during I-Q cycle. If this light is on and the PRIV OP is off, the check is caused by an invalid Q-byte in an I/O instruction.

Figure 4-9 (Part 3 of 3). Information Displayed on Legend Strips

MACHINE CYCLES—All Models

Twelve back-lit indicator lamps (Figures 4-6, 4-7, and 4-8) represent the 12 mutually exclusive machine cycles. In the process mode, they identify the cycle in progress. In the step mode, they identify the cycle either in progress or just completed. The I/O CYCLE light is on during the test mode of operation. No MACHINE CYCLE indicator is on after a system reset, after a STOP key was pressed, or during an address compare stop. INT LEV at the right end of the machine cycles lamps is described separately in this section.

INT LEV (Interrupt Level)—All Models

This back-lit indicator (Figures 4-6, 4-7, and 4-8) comes on when the processing unit is servicing any of the interrupt levels. The Model 12C CPU is servicing interrupt level 0 if the INT LEV light is on and INT 1, INT 2, INT 3, and INT 4 lights are off. The Model 15 CPU is servicing interrupt level 0 if the INT LEV light is on and INT 1, INT 2, and INT 4 lights are off.

INT 1, INT 2 and INT 4—Models 15C and 15D Only

These back-lit lights, located below the left end of the MACHINE CYCLE indicator strip (Figures 4-7 and 4-8) define which interrupt level the processing unit is servicing whenever the INT LEV light is also on. If INT LEV is on, but INT 1, INT 2, or INT 4 is not on, the processing unit is servicing interrupt level 0; otherwise, the processing unit is servicing the interrupt level by adding the binary values of the INT 1, INT 2, and INT 4 lights that are on with the INT LEV light. For example, INT 2 light on but INT 1 and INT 4 lights off indicates the processing unit is servicing interrupt level 2; if INT 1, INT 2, and INT 4 are all on, the processing unit is servicing interrupt level 7.

CLOCK (Back-Lit Indicators)—All Models

Ten indicator lamps (Figures 4-6, 4-7, and 4-8) represent clocks 0 through 9, which can be stepped through in the CE clock-step mode. In the normal process mode, a machine cycle consists of clocks 0 through 8, inclusive. Clock 9 is used with the CE step and test modes.

TH CHK (Thermal Check)—All Models

This back-lit light (Figures 4-6, 4-7, and 4-8) turns on whenever one of the system thermal sensors (located in the processor and in the line printer) detects an overheated condition. If this condition occurs, the processing unit removes power from the system. (The PWR CHK light also comes on, remaining on until the POWER switch is moved to the off position.) The TH CHK light remains on until the overheated condition has been corrected and the POWER switch has been turned off. Power can then be restored to the system by turning the POWER switch on.

Figure 4-10 summarizes power check/thermal indications and the required action.

PWR CHK (Power Check)—All Models

PWR CHK (Figures 4-6, 4-7, and 4-8) lights whenever the POWER switch is on and power is not completely applied to the system, or whenever the POWER switch is off and power is not completely removed from the system (except in those areas within the power control circuitry where power is never completely removed). The following statements apply to PWR CHK light operation:

- When the POWER switch is turned on, the PWR CHK light remains on until power has sequenced all the way up and the system is ready to operate.
- When the POWER switch is turned off, the PWR CHK light remains on until power has sequenced all the way down.
- If system power is on and is then removed from the system because an over temperature condition has been detected (see *TH CHK*), the PWR CHK light remains on until the POWER switch is turned off.
- If system power is on and is then removed from the system because a power fault has been detected, the PWR CHK light remains on until the POWER switch is turned off.

After the power fault has been corrected, power is restored to the system by placing the POWER switch in the off position, pressing the CHECK RESET key, then turning the POWER switch to the on position.

Note: Although the IBM 5445 power can be controlled remotely by the processing unit POWER switch, 5445 power is not included in the power check indication.

LAMP TEST Key—All Models

Pressing this key (Figures 4-6, 4-7, and 4-8) turns on all the processing unit display lights.

Fault	POWER ON/ OFF Switch	Indicators		Action
		PWR CHK	TH CHK	
Internal power supply malfunction or (Model 12 only) 3340 power supply malfunction	On	On	Off	1. Turn POWER switch to off. 2. Call CE. 3. Correct problem. 4. Press CHECK RESET. 5. Turn power on.
Thermal condition	On	On	On	1. Turn POWER switch to off. PWRK CHK indicator goes off, TH CHK light stays on until condition is removed. 2. Call CE.
Customer power source loss	On	On	On	1. Turn POWER switch to off. 2. All indicators turn off. 3. Turn POWER switch to on and continue operation.
Emergency power off (EPO) activated	On	Off	Off	1. Turn POWER switch to off. 2. Call CE. 3. Correct problem. 4. Restore EPO interlock. 5. Turn POWER switch to on.

Figure 4-10. Power Check/Thermal Indicators and Action

COMMUNICATIONS ADAPTER OPERATOR PANEL

xxxx ATTN/yyyy ATTN (Attention) Light

For specific attention lights on the various models, see Note 2 of Figure 4-11. The following table shows the conditions indicated by these two lights:

Instruction	Condition Indicated
Any receive or transmit and receive or (on non-switched and multi-point networks only) receive initial	Data set not ready
Auto call or receive initial on switched network	Auto call unit power off or data line being used
Any SIO except control SIO	Either BSCA disabled or external test switch on and BSCA not in test mode
None	Data set not ready

TSM MODE (Transmit Mode) Light

The TSM MODE light (Figure 4-11) indicates that the adapter has been instructed to perform a transmit operation.

RECEIVE MODE Light

This light (Figure 4-11) indicates that the adapter has been instructed to perform a receive operation.

RECEIVE INITIAL Light

This light (Figure 4-11) is turned on by an SIO receive initial instruction. It is turned off at the end of the receive initial operation.

CONTROL MODE Light

This indicator (Figure 4-11) is used only on systems that have the station select feature installed. The light is turned on by an EOT sequence during a transmit, receive, or receive initial monitor operation when the station select feature is installed. It is turned off by the decoding of an SOH or STX.

ACU PWR OFF (Auto Call Unit Power Off) Light

The ACU PWR OFF light (Figure 4-11) indicates that the auto call unit (special feature) power is off.

DT TERM READY (Data Terminal Ready) Light

The DT TERM READY light (Figure 4-11) indicates that the BSCA is enabled and that the data terminal is ready for use.

TEST MODE Light

This light (Figure 4-11) indicates that the program has placed the adapter in a test mode of operation.

CLEAR TO SEND Light

This light (Figure 4-11) indicates that the clear to send line from the data set is on and that the adapter may now transmit.

CHAR PHASE (Character Phase) Light

The CHAR PHASE light (Figure 4-11) indicates that the adapter has established character synchronism with the transmitting station. The light is turned off at the end of receive operations and whenever character synchronism is lost.

BUSY Light

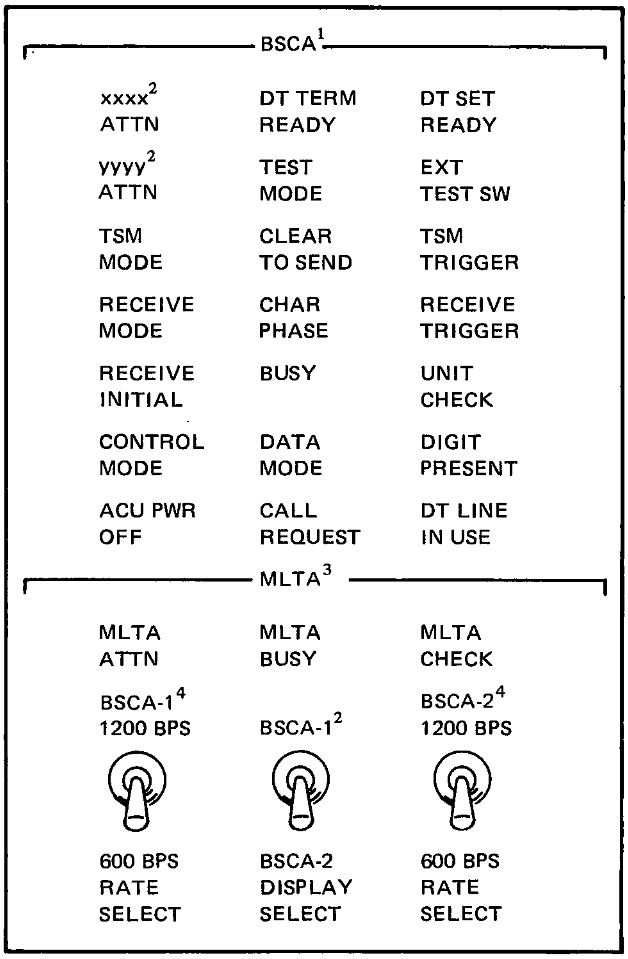
This light (Figure 4-11) indicates that the communication adapter is executing a receive initial, transmit and receive, auto call, receive or loop test instruction.

DATA MODE Light

This light (Figure 4-11) is turned on by the decoding of an SOH or STX during a transmit or a receive operation. It is turned off at the end of the transmit or receive operation.

CALL REQUEST Light

On systems with the auto call feature installed, this light (Figure 4-11) indicates that the communication adapter has received an SIO auto call instruction and is performing an auto call operation.



[1]This heading varies, depending on the features installed.
[2]These lights read:

	5408	*5412*	*5410/5415*
xxxx –	ICA	BSCA BSCA-1	BSCA BSCA-1 LCA
yyyy –	BSCA	BSCA-2 ICA	BSCA-2

[3]MLTA is available by RPQ only.
[4]Rate select switch is for machines used outside the U.S.A. If the rate selection feature is specified on either of the BSCAs, it will be made available to both.

Figure 4-11. Typical Communications Control Panel

DT SET READY (Data Set Ready) Light

The DT SET READY light (Figure 4-11) indicates that the data set ready line from the data set is on and that the data set is ready for use. If the BSCA is equipped with an EIA Local Attachment feature, this light indicates that the attached device is ready.

EXT TEST SW (External Test Switch) Light

The EXT TEST SW light (Figure 4-11) indicates that the switch at the data set end of the medium speed data set cable is in the test position. For high-speed data sets and the 1200 BPS integrated modem feature, this indicator is active when the local test switch on the CE panel is in the on position.

TSM TRIGGER (Transmit Trigger) Light

The TSM TRIGGER light (Figure 4-11) indicates the status of the transmit trigger. The light is on when the trigger is at a binary 0 state.

RECEIVE TRIGGER Light

This light (Figure 4-11) indicates the status of the receive trigger. The light is on when the trigger is at a binary 0 state.

UNIT CHECK Light

This light (Figure 4-11) turns on when any bit in status byte 2 is on. Also, when an SNS transition or SNS stop register instruction is executed, it is possible for an LSR, S-register, or DBI register parity check to occur, resulting in a unit check condition with the UNIT CHECK light on. Under such a condition, the status byte 2 bits may all be 0.

The unit check indicator signifies that the BSCA program should enter an error recovery procedure.

DIGIT PRESENT Light

This light (Figure 4-11) indicates that a digit was obtained from storage for the auto call unit when the auto call feature was installed.

DT LINE IN USE (Data Line in Use) Light

On systems with the auto call unit installed, the DT LINE IN USE light (Figure 4-11) indicates that the data line occupied line from the auto call unit is on.

RATE SELECT Switch

This switch (Figure 4-11) which is present only on systems installed outside the USA that have the rate selection feature as well, controls the rate of transmission and reception of data.

CE CONTROLS FOR BSCA—ALL MODELS

CE control switches should be altered only when the system is stopped.

CABLE TEST Switch—All Models

This switch is part of the plug at the remote end of the BSCA data cable; that is, at the data set end of the cable. The switch should be set at the operate setting except during BSCA diagnostic operations. This switch is provided with data cables to medium speed data sets only.

BSCC OPERATOR PANEL

BSCC ATTN Light

The BSCC ATTN light (Figure 4-12) turns on whenever the BSCC turns on the system I/O ATTENTION light.

DT TERM READY Light

The DT TERM READY light (Figure 4-12) indicates the enable or disable status of the BSCC. This condition occurs when the attachment is enabled and the microcontroller has the microcode loaded.

DT SET READY Light

The DT SET READY light (Figure 4-12) indicates that the data set ready line from the data set is on and that the data set is ready for use. If the BSCC is equipped with an EIA Local Attachment feature, this light indicates that the attached device is ready.

SEND/RCV DATA Light

The SEND/RCV DATA light (Figure 4-12) indicates a binary 1 is being transmitted or received. This light is for diagnostic use.

TEST MODE Light

The TEST MODE light (Figure 4-12) indicates the program has placed the BSCC in test mode.

EXT TEST SW Light

The EXT TEST SW light (Figure 4-12) indicates the test switch at the end of the medium speed cable is in the test position or the 'Test Control' latch is set (causes data wrap on the BSCC board).

TSM MODE Light

The TSM MODE light (Figure 4-12) indicates the BSCC was instructed to perform a transmit operation on the selected line.

CLEAR TO SEND Light

The CLEAR TO SEND light (Figure 4-12) indicates the clear to send signal from the data set for the selected line is active and the BSCC is free to transmit on the line.

BUSY Light

The BUSY light indicates that a line is busy as a result of processing a functional SIO command.

RECEIVE MODE Light

The RECEIVE MODE light (Figure 4-12) indicates the BSCC was instructed by the program to perform a receive instruction on the selected line.

RECEIVE INITIAL Light

The RECEIVE INITIAL light (Figure 4-12) indicates the BSCC was instructed by the program to assume a receive initial mode and wait for information to be received on the selected line.

UNIT CHECK Light

The UNIT CHECK light (Figure 4-12) indicates the BSCC has an I/O check condition and cannot continue until it is corrected.

DISPLAY SELECT Switch

This switch (Figure 4-12), which is present only when the BSCC line 2 feature is installed, allows the operator to select one of the two BSCC lines for display.

Note: This switch should be altered only when the system is stopped.

RATE SELECT Switches

These switches (Figure 4-12), which are present only when the rate select and BSCC line 2 features are installed, allows the operator to select full or half rate clocking speeds.

Note: These switches should be altered only when the system is stopped.

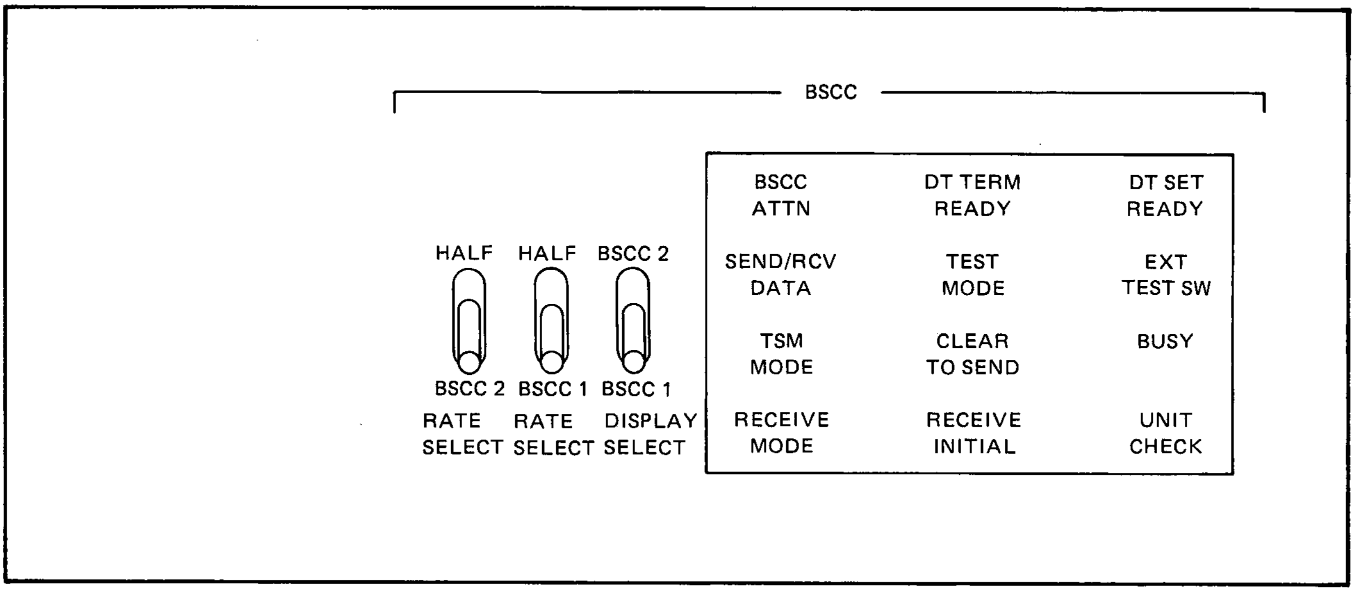


Figure 4-12. BSCC Control Panel

CE CONTROLS FOR BSCC–ALL MODELS

CE control switches should be altered only when the system is stopped.

CABLE TEST Switch–All Models

This switch is part of the plug at the remote end of the BSCC data cable; that is, at the data set end of the cable. The switch should be set at the operate setting except during BSCC diagnostic operations. This switch is provided with data cables to medium speed data sets only.

DUAL PROGRAM CONTROL PANEL

Figure 4-13 shows the Dual Program Feature control panel.

Message Display Units

A message display unit is provided for each program level. These units operate in the same manner as the message display unit in the system controls.

PROCESS Lights

These lights indicate which program level is functioning at any time. If an interrupt is being serviced, this indicator shows which index registers and program status register are in use.

HALT RESET Keys

These keys are used to take a program level out of the programmed halt state. Pressing either of these keys clears the corresponding message display unit and allows the corresponding program to continue its normal operation.

INTER Key/Light

Pressing this key when it is illuminated causes the program in operation at that time to halt its normal operation and enter the interrupt-handling subroutine for interrupt level 0. Normal programmed operation will be resumed after the interrupt routine signals completion of interrupt servicing with a start I/O instruction to reset interrupt request 0.

The interrupt key/light is on only when the system is in dual program mode and interrupt level 0 is enabled. Selection of whether the system is to be used in the dedicated or the dual program mode is accomplished via the start I/O instruction. The start I/O instruction is also used to enable or disable the use of interrupt level 0.

DUAL PROGRAM CONTROL Switch

This rotary switch is normally used in conjunction with the console interrupt key. The status of this switch is checked by the test-I/O-and-branch instruction.

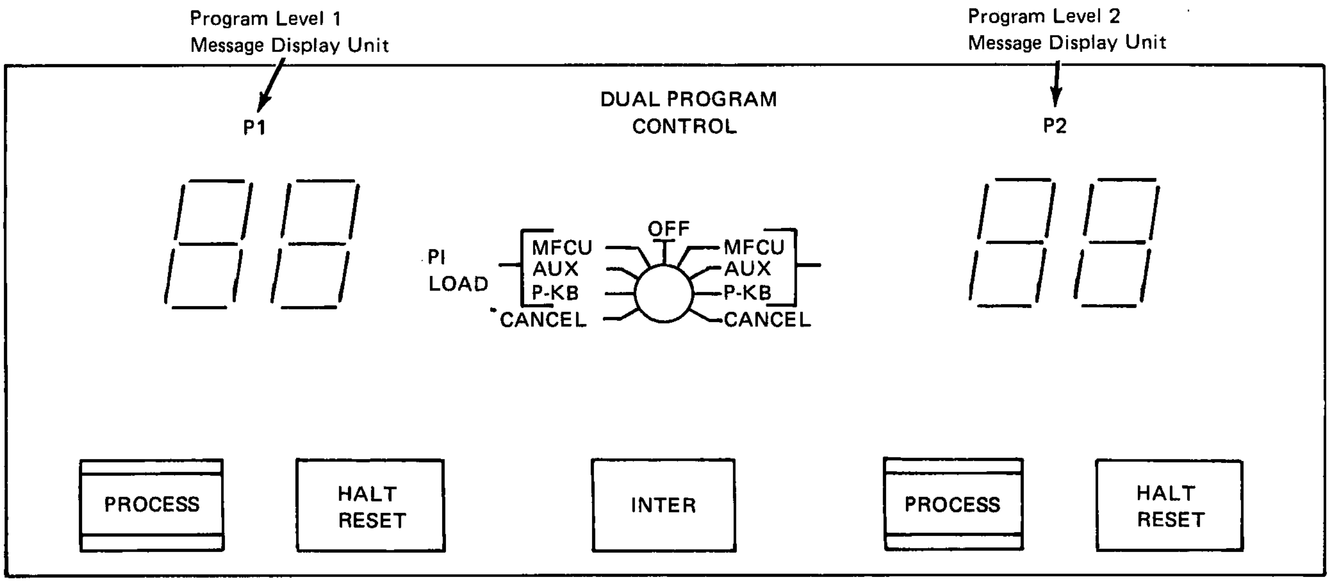


Figure 4-13. Dual Program Control Panel

CE CONTROLS

Figures 4-14 through 4-18 illustrate the CE control panels used on various IBM System/3 models.

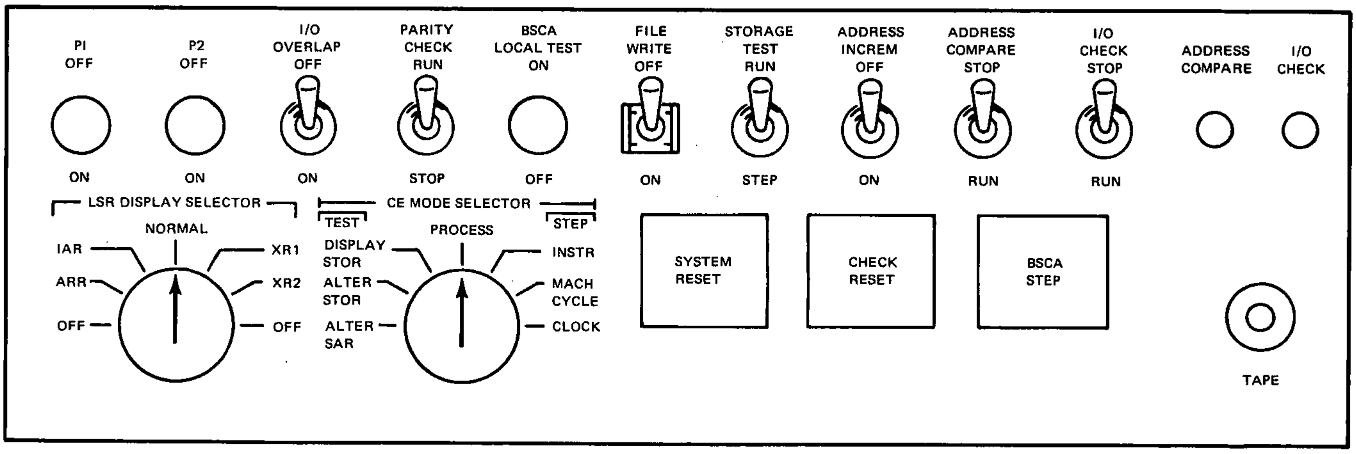
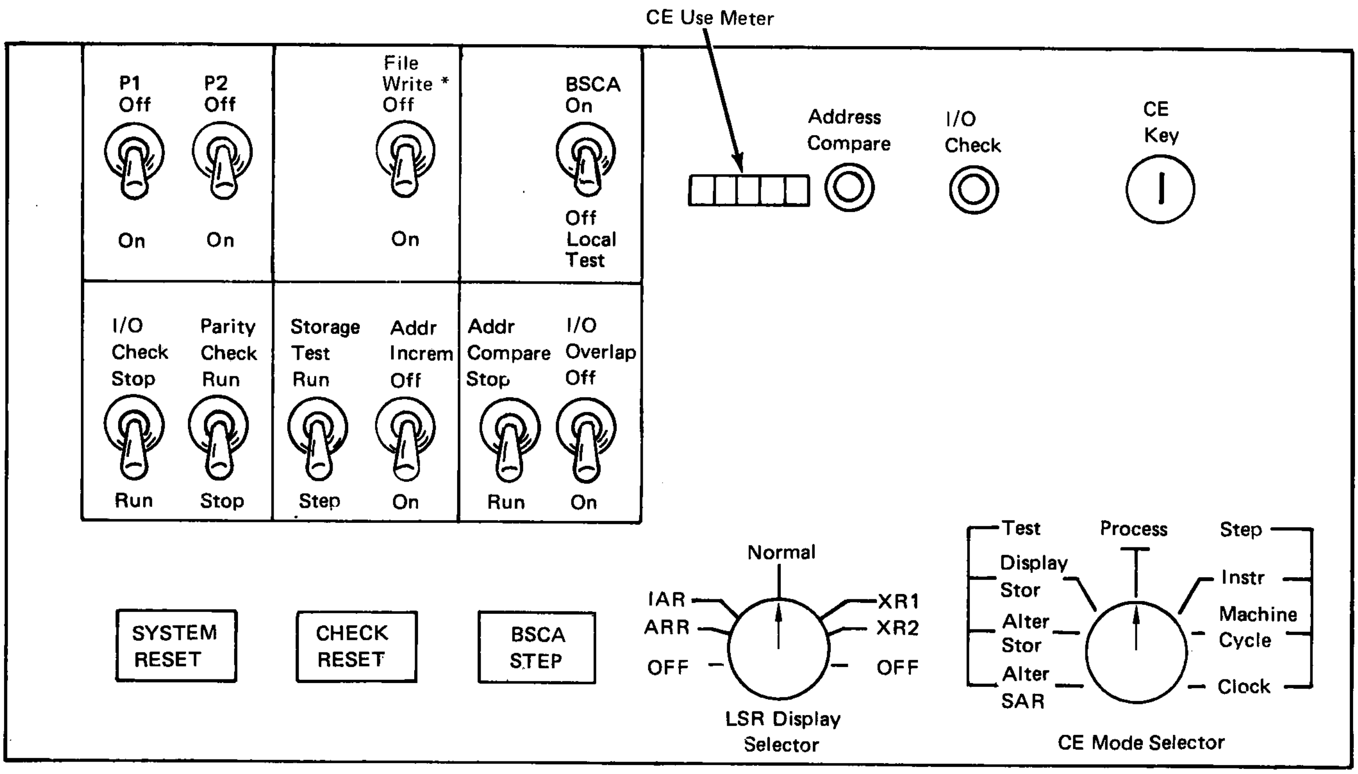
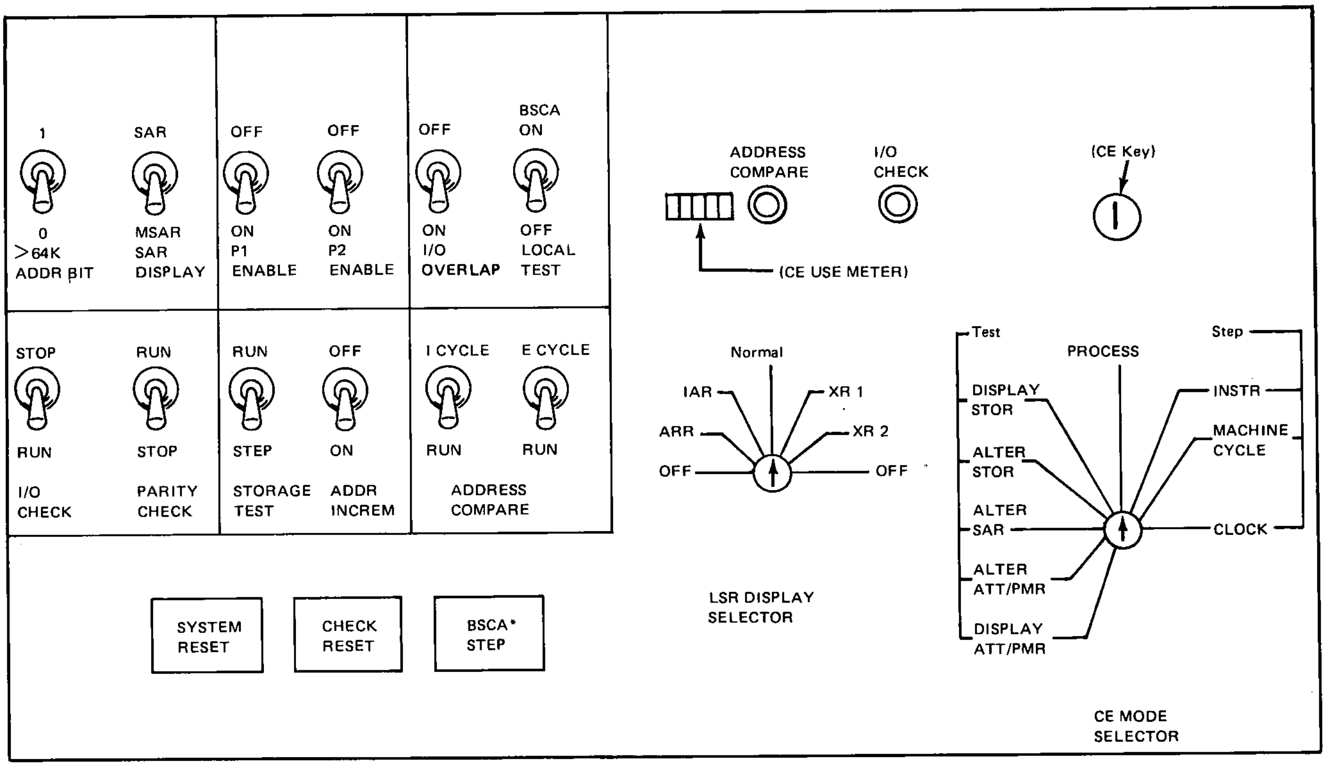


Figure 4-14. CE Control Panel on Model 8



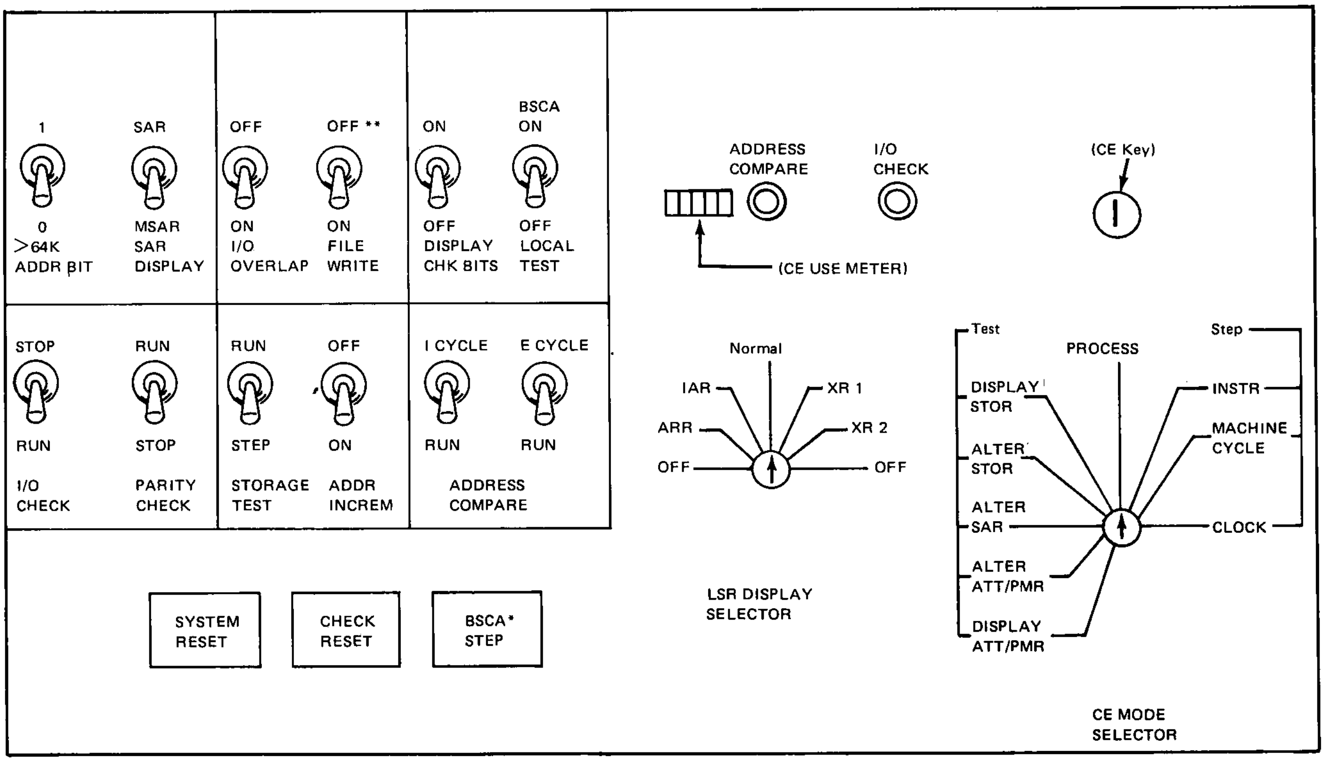
*Not on a Model 12

Figure 4-15. CE Control Panel on Models 10 and 12



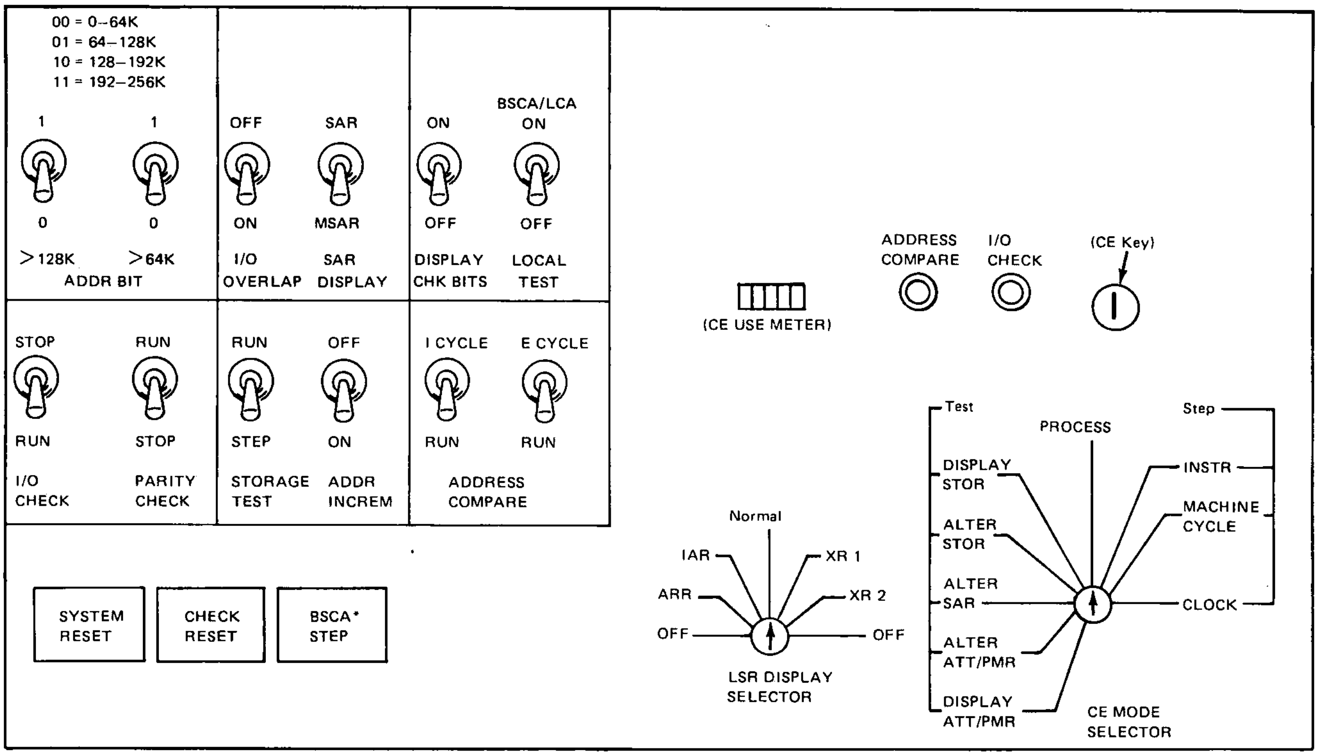
*If local communications adapter feature is installed, this switch is labeled LCA/BSCA STEP.

Figure 4-16. CE Control Panel on Model 12C



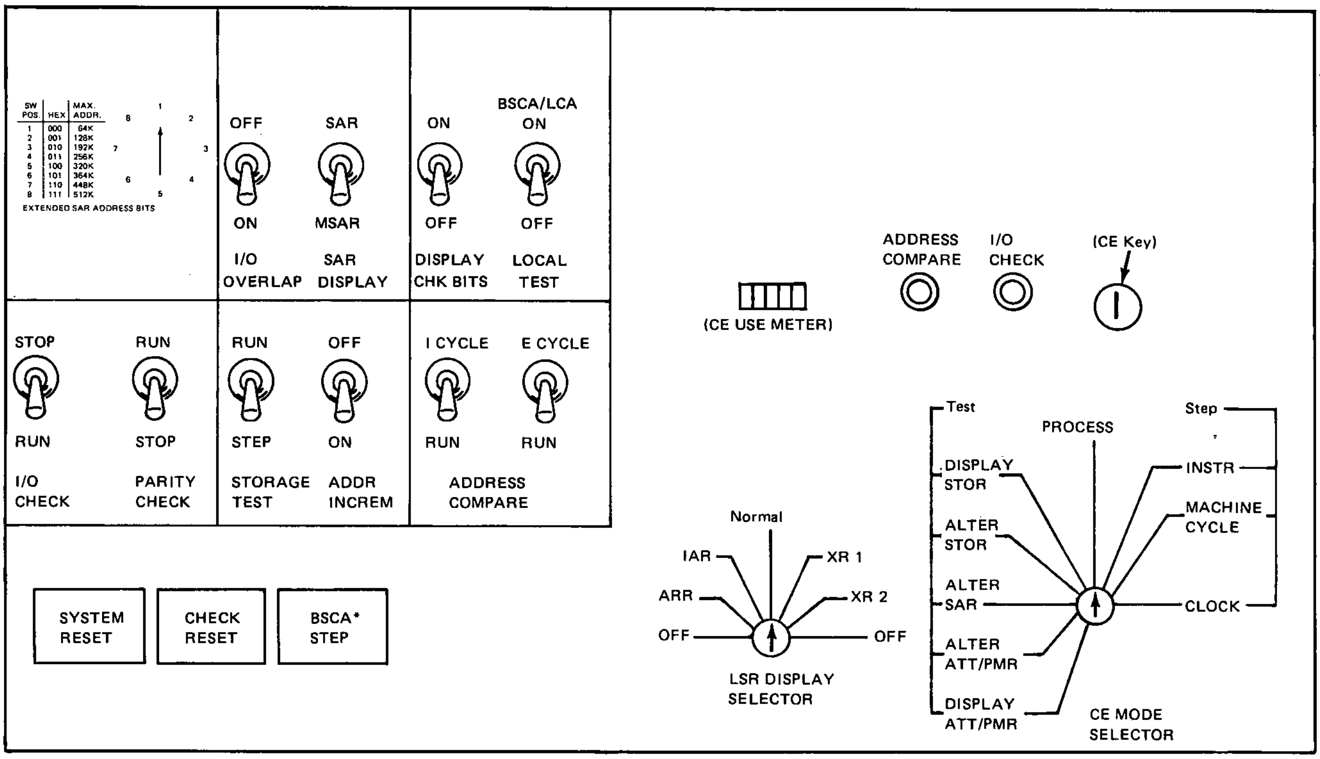
*If local communications adapter feature is installed, this switch is labeled LCA/BSCA STEP.
**Not on Model 15B.

Figure 4-17. CE Control Panel on Models 15A and 15B



*If local communications adapter feature is installed, this switch is labeled LCA/BSCA STEP.

Figure 4-18. CE Control Panel on Models 15C and 15D (Except Models D25 and D26)



*If local communications adapter feature is installed, this switch is labeled LCA/BSCA STEP.

Figure 4-19. CE Control Panel on Model 15 D25 and Model 15 D26

CE KEY Switch—Models 10, 12, and 15

This switch (Figures 4-15 through 4-19) is operated by the CE to prevent recording time when the system is being serviced.

CE MODE SELECTOR—All Models

This rotary switch (Figures 4-14 through 4-19) selects one of three processing unit operating modes: normal PROCESS mode, STEP mode, or TEST mode. PROCESS is the normal mode for normal programmed system operation.

In the STEP mode, the rotary switch setting controls the manner in which the processing unit performs the stored program:

1. *INSTR (Instruction) STEP:* Each time you press and release the START key the processing unit performs one complete instruction. The I-phase of the instruction occurs when you press START. If the instruction has an E-cycle, the E-phase occurs when you release START.

   Note: When you press START for a start I/O instruction, this instruction is completely executed. Then the next sequential instruction is also executed.

2. *MACHINE CYCLE STEP:* Each time you press and release the START key the processing unit advances the instruction through one machine cycle. When you press START, the processing unit accesses 1 byte of data in storage, modifies it as required, and displays the result in the ALU (arithmetic and logical unit) indicators of the console display. When you release START, the processing unit stores either the old data or the new result (depending on the operation being performed) back in the storage position from which the byte was accessed.

3. *CLOCK STEP:* Each time you press START the processing unit takes an odd-numbered clock cycle; when you release START the processing unit takes the next sequential (even-numbered) clock cycle.

   Note: The clock advances automatically from I-phase end of every executable start I/O instruction until data transfer is complete. This ensures data integrity during I/O data transfer. In CLOCK STEP mode, the START key is not functional during I/O data transfer.

   In CLOCK STEP mode, the HALT IDENTIFIER lights are not turned on for any of the steps above.

The switch settings under TEST, and the associated CPU functions are:

1. *ALTER SAR:* The processing unit loads the address set up on the four rotary console switches and (Model 15 only) the CE address switch(es) on the CE panel into storage address register (SAR) when you press START. At the same time, the processing unit loads the address set up in the four rotary console switches (but not the data set into the CE address switch(es)) into the instruction address register (IAR). The CE address switch bits are not stored in the IAR, which works with logical 16-bit addresses only.

   When you alter the storage address register on the Models 12C and 15, to alter or display the contents of storage, you must enter the 17-bit, 18-bit, or 19-bit (Model 15 only) address of the storage location. This is required because the processing unit automatically disables the address translate table (ATT) registers in alter SAR mode, and the 17-bit, 18-bit, or 19-bit address entered from the console and CE panel switches addresses storage untranslated.

   When you alter the storage address register on the Models 12C and 15, to manually branch to a routine, you must enter the 16-bit *logical* address. You must also enter the *logical* address to restart a program. The logical address must be used in these cases because the bit values in the CE address switches are ignored in PROCESS mode and STEP mode, and these bits are essential to specify physical addresses greater than 64K.

2. *ALTER STOR (storage):* Pressing START loads data set up in the data switches (the rightmost two rotary switches on the console) into the A-register. Releasing START then transfers that data from the A-register into the storage position specified by SAR and into the Q-register. The Models 12C and 15 ATT registers are inactive during ALTER STOR mode operations, and the address used is the 16 bits from the IAR and the CE address switch values entered into the processing unit by the most recent alter SAR operation.

3. *DISPLAY STOR (storage):* When you press START, the processing unit transfers data from the storage position specified by SAR into the B-register. Then, when you release START, the processing unit loads the data in the B-register into the Q-register. The Models 12C and 15 ATT registers are not active in DISPLAY STOR mode, so the address used is the 16 bits from the IAR and the CE address switch values entered into the processing unit by the most recent alter SAR operation. To display the contents of storage, turn the display panel roller to strip 3 and read the display lights from the Q-register on the display strip.

4. *ALTER ATT/PMR (Models 12C and 15):* This mode of operation lets you alter the contents of the various ATT registers and program mode registers, one at a time. To alter the contents of the registers:

   a. Select the register to be loaded by entering the ATT register number or the PMR identification number into rotary switches 1 and 2, the two leftmost rotary switches on the console panel. (The appropriate switch settings are shown in Figure 4-20.)

   b. Enter the data to be stored in the register into rotary switches 3 and 4, the two rightmost rotary switches on the console panel. Set switch 3 at the position representing the first hex digit in the byte, and switch 4 at the position representing the second hex digit in the byte. When altering the I/O > 256K bit, an odd value in switch 4 turns the bit on and an even value turns the bit off.

   c. Press START. When you press START, the processing unit transfers the data from the data switches (rotary switches 3 and 4) into the A-register. When you release the key, the CPU transfers the data from the A-register into the register ATT or PMR specified by rotary switches 1 and 2.

5. *DISPLAY ATT/PMR:* This position (Models 12C and 15) lets you display the contents of one of the ATT or PMR registers. To do this:

   a. Select the register whose content is to be displayed by entering the ATT register number or the PMR identification number into rotary switches 1 and 2, the two leftmost rotary switches on the console panel. The appropriate switch settings are shown in Figure 4-20.

   b. Turn the console display roller to position 6 if you want to display the contents of an ATT register, or to position 7 if you want to display the contents of a PMR.

   c. Press START. If you have selected an ATT register, the CPU displays the register contents. If you have selected a PMR, the processing unit loads the address into the Q-register, then displays the register contents in the display strip lights.

The storage test switch must be in the step position to avoid a processor check when the CE MODE SELECTOR switch is moved between the alter storage position and the display storage position.

Note: No test is made for invalid storage addresses when the CE MODE SELECTOR switch is in one of the test positions.

	ATT/PMR Address Switch Settings	
Register to Be Loaded	**Switch 1[4]**	**Switch 2**
Att Register XX[1]	*	*
Program Level 1 PMR	2	0
Program Level 2 PMR[2]	2	1
Interrupt Level 0 PMR	2	8
Interrupt Level 1 PMR	2	9
Interrupt Level 2 PMR	2	A
Interrupt Level 3 PMR	2	B
Interrupt Level 4 PMR	2	C
Interrupt Level 5 PMR[3]	2	D
Interrupt Level 6 PMR[3]	2	E
Interrupt Level 7 PMR[3]	2	F

[1] Enter the first digit of the ATT register number into switch 1, and the second digit of the ATT register number into switch 2 to identify the desired ATT register. ATT registers are numbered sequentially in hex from 00 to 1F.

[2] Model 12C only.

[3] These interrupt levels are not used on the Model 12C.

[4] Set this switch to 3 to alter the I/O > 256K PMR bit.

Note: Settings not defined above may cause undefined errors.

Figure 4-20. ATT/PMR Register Address

LSR DISPLAY SELECTOR—All Models

This rotary switch (Figures 4-14 through 4-19) lets you display the contents of a local storage register. This switch is operative whenever the processor clock is stopped, or if the clock is running, when no processing unit machine cycle or I/O data transfer cycle is being taken. Otherwise, the system controls the display of the LSRs.

The switch has these positions: NORMAL, IAR, ARR, XR1, XR2, and OFF.

With the switch set at NORMAL, the system controls the selection and display of the LSRs. The OFF position is provided for CE use.

When the switch is set at IAR, ARR, XR1 or XR2, the specified LSR for the program or interrupt level in use is selected and its contents are available for display. The displayed content of the switch-selected LSR will be erroneous if an I/O LSR is being selected at the time of display.

Note: LSR parity check display will reflect the parity of the switch-selected LSR. Therefore, if the selector switch is not kept in the normal position during normal processing, LSR parity checks will not show in the LSR processor check display (strip 8), or the LSR display may show erroneous information.

SYSTEM RESET Key—All Models

This key (Figures 3-11 through 3-14) initiates a system reset if you press it while the CE MODE SELECTOR switch is set at the PROCESS setting. At all other settings, SYSTEM RESET is inoperative.

A system reset puts the system in an immediate idle state. Processing unit registers, controls, and status indicators are reset (unless otherwise specified in this manual). A complete program restart is normally required after a system reset.

On Models 12 and 15, the first time you press the SYSTEM RESET key or perform an IPL operation after a power on sequence, the processing unit loads the data specified by the rightmost two rotary console (DATA) switches into each position of main storage. (This action is called *initial memory scan* and does not occur unless the mode switch is set to PROCESS when the power on sequence is performed.)

A system reset operation on a Model 8, 10, or 12 resets the program level 1 instruction address register (P1IAR) and program levels 1 and 2 program status registers (P1PSR and P2PSR) to 0.

A system reset on Model 15 resets the program level instruction address register (P-IAR) and program level program status register (P-PSR) to 0, and disables the program mode register control over the CPU. With PMR control disabled, the PMR translation, storage protection, I/O > 64K, I/O > 128K, I/O > 256K, and interrupt masking functions are all inactive. However, the processing unit is in privileged mode following the reset.

CHECK RESET Key--All Models

Pressing this key (Figures 4-14 through 4-19) resets the processor check, input/output check, and system power check conditions and allows a power on retry. A check reset removes the current error conditions, allowing the processing unit to resume its operation when you press START.

Note: The CHECK RESET key immediately resets all 3411 and 5445 functions and status indicators. Therefore, do not press CHECK RESET while the 3411 or 5445 attachment is processing I/O instructions.

BSCA Step Key—Models Using BSCA

The BSCA STEP key, which is effective only when the communication adapter is in step mode, causes the communication adapter to advance 1 bit each time a key is used. (This key is labeled ICA/BSCA STEP if the integrated communications adapter feature is installed.)

CE Servicing Switches

The following switches are used only by the customer engineer. (Some models do not have all these items):

- I/O OVERLAP
- I/O CHECK
- PARITY CHECK
- BSCA LOCAL TEST
- BSCA/LCA LOCAL TEST
- DISPLAY CHK BITS
- TAPE Jack

ADDRESS COMPARE Light—All Models

This light (Figures 4-14 through 4-19) comes on when the processing unit stops because it has detected the address specified for an address compare function (see *ADDRESS COMPARE Switch* and *SAR DISPLAY Light*).

I/O CHECK Light—All Models

This indicator (Figures 4-14 through 4-19) turns on when certain I/O errors are detected by an I/O device. It is turned off by a system reset operation, a check reset operation, or by the I/O device itself.

STORAGE TEST Switch—All Models

With STORAGE TEST (Figures 4-14 through 4-19) set at the STEP position, the processing unit accesses the storage location specified by the storage address register once each time someone presses the START key. With STORAGE TEST set at the RUN position, pressing START repetitively either accesses the same position of storage repeatedly or accesses positions in a 64K segment of storage sequentially.

To use the lower 64K of storage on the Models 12C and 15, first execute an ALTER SAR cycle with the CE address switches set to 0 or perform a system reset operation. To use main storage above 64K on the Models 12C and 15, first execute an ALTER SAR cycle with the appropriate ADDR BIT switch setting. After setting SAR as described above, you can start the storage test operation.

Note: The STORAGE TEST switch must be in the STEP position to avoid a processor check when changing the CE MODE SELECTOR from ALTER STORAGE position to DISPLAY STORAGE position and vice versa.

ADDR INCREM (Address Increment) Switch—All Models

This switch (Figures 4-14 through 4-19) controls the contents of the SAR (storage address register) while the CE MODE SELECTOR switch is set at ALTER STOR or DISPLAY STOR. With ADDR INCREM at the ON position, the processing unit adds a value of 1 to the address in the SAR after each storage access cycle. With ADDR INCREM at the OFF position, the address in SAR remains unchanged at the end of each storage access cycle.

Note: On the Model 15, the processing unit does not advance the SAR address from one 64K block of storage to the next higher 64K block of storage; instead, the processing unit advances the SAR address to hex 0000 and the address starts again at the beginning of the 64K specified by the settings of the CE address toggle switch(es). To cross from any 64K block to the next higher 64K block, you must select the next higher 64K block by setting the CE address switch(es) to their appropriate positions, setting 0000 into the four rotary address switches, then performing an alter storage address operation.

ADDRESS COMPARE Switch—Models 8, 10, and 12B

The ADDRESS COMPARE switch (Figures 4-14 and 4-15) in conjunction with the SAR DISPLAY switch, is used to stop program execution at desired main storage addresses when the system is operating in PROCESS mode.

With SAR DISPLAY in SAR position, the system operating in PROCESS mode, the register display roller switch set to strip 1 (SAR HI/SAR LO), and the ADDRESS COMPARE switch set at STOP, the processing unit continues to compare the current main storage address to the address set in the four console address switches. A match of the console address switches and the SAR display stops the processor at the end of the storage read/write cycle during which the address match occurred. The integrity of I/O data transfers is preserved during address compare stop operations. To restart the processor after an address compare stop, press the START key.

ADDRESS COMPARE Switches—Models 12C and 15

The two ADDRESS COMPARE switches, (Figures 4-16 through 4-19) in conjunction with the SAR DISPLAY switch, provide the capability to stop on I/O-cycles or I-cycles and/or E-cycles of either a real or a logical address. This happens when the system is operating in PROCESS mode, and the register display roller switch is set to strip 1 (SAR HI/SAR LO). Both the compare switches should be set at RUN if you do not want to stop the program.

With SAR DISPLAY in SAR position, the processing unit continually compares the 16-bit logical address in SAR to the bits in the four hex-digit address set in the four console address switches. When SAR DISPLAY is in MSAR position, the processing unit continues to compare the real address (17-bit address applied to Model 12C, 15A, or 15B main storage, 18-bit address applied to Models 15C and 15D main storage or 19-bit address applied to Model 15 D25 or Model 15 D26 main storage) to the address set in the four console address switches and the ADDR BIT switches on the CE panel.

With the I-CYCLE switch on and the E-CYCLE switch off, an address compare stop occurs on an I-cycle only.

With the E-CYCLE switch on and the I-CYCLE switch off, an address compare stop occurs on either an E-cycle or an I/O cycle.

With the ADDRESS COMPARE switches at I-cycle and E-cycle positions, an address compare stop occurs whenever the processing unit detects a compare equal condition.

During address compare stop mode processing, a match of the console data switches and the register display stops processor at the end of the storage read/write cycle. To restart the processor, press the START key.

Note: The integrity of I/O data transfers is preserved. The contents of SAR do not necessarily match the setting of the address switches at stop time. If a match occurs on a physical address the processing unit displays the logical address in the SAR display when the processing unit stops.

ADDR BIT and EXTENDED SAR ADDRESS BITS Switches—Models 12C and 15

>64K ADDR BIT (Models 12C, 15A, 15B, 15C, and 15D): This switch (Figures 4-17, 4-18, and 4-19) enters a 64-bit into the SAR for use in addressing storage positions between 65,536 and 131,072, or above 196,608 (decimal) in binary to display storage, alter storage, or perform address compare operations. System reset sets SAR to 0. This switch is inoperative in PROCESS mode except for address compare stop operations.

>128K ADDR BIT (Models 15C and 15D): This switch (Figure 4-18) enters a 128-bit into the SAR for use in addressing storage positions above 131,072 (decimal) in binary to display storage, alter storage, or perform address compare operations. System reset sets SAR to 0. This switch is inoperative in PROCESS mode except for address compare stop operations.

EXTENDED SAR ADDRESS BITS (Models 15 D25 and D26): This rotary switch (Figure 4-18) enters the bit represented by the switch position into the SAR for use in addressing the various storage positions. This allows the operator to display storage, alter storage, or perform address compare operations. System reset sets SAR to 0. This switch is inactive in PROCESS mode except for address compare stop operations.

SAR DISPLAY Switch—Models 12C and 15

This switch (Figures 4-16 through 4-19) controls the display in roller position 1. In the SAR position, the display reflects the untranslated or logical address (only 16 bits of SAR are ever displayed in this position).

In the MSAR position, the display reflects the actual address sent to memory. If translation is active, the SAR HI bits come from the ATT register. During I/O cycles, the SAR HI bits come from I/O local storage registers. During console operations, the SAR HI bits come from the > 64K ADDR BIT switch, (on the Models 15C and 15D) the > 128K ADDR BIT switch and the EXTENDED SAR ADDRESS BITS switch.

When this switch is used with the ADDRESS COMPARE stop switches, it controls whether the stop occurs on logical (16-bit) or real (17-bit on Models 12C, 15A, and 15B; 18-bit on Models 15C and 15D; 19-bit on Model 15 D25 and Model 15 D26) addresses.

This switch is ineffective in TEST mode and at certain times in PROCESS mode. If the system is stopped in PROCESS mode and no I- or E-cycle light is on, then the SAR DISPLAY switch is ineffective. (This is because the address translator is only active during I, EA, or EB cycles.) I-cycles include I-OP, I-Q, I-HI, I-L1, I-L2, I-X1, I-X2. If the system is stopped with no I-cycle or E-cycle active the logical and read addresses will appear to be equal. If the system stops and an I-cycle or E-cycle light is on, the switch is effective. If the real address and logical address are equal in this situation, the program is not using the address translator.

FILE WRITE Switch—Models 8, 10, and 15A

This switch (Figures 4-14 through 4-17) when set at its OFF position, prevents the processing unit from writing to disk.

MANUAL OPERATION PROCEDURES

Altering Storage Data

1. Press STOP.
2. Set the STORAGE TEST switch to STEP.
3. Set the CE MODE SELECTOR switch to ALTER SAR.
4. Set the ADDRESS/DATA switches on the console to the address of the storage position holding data to be altered. For Models 12C and 15, set both the ADDRESS/DATA switches on the console and the CE address switch(es) on the CE panel to the address of the storage position holding data to be altered.
5. Press START (switch) on the system control panel.
6. Turn the CE MODE SELECTOR switch to ALTER STOR.
7. Set the two rightmost address/data switches to the hex value you want in storage.
8. Press START.

In order to resume normal operation it will be necessary to set the storage address register to the address of the instruction with which you wish to begin.

Displaying Storage Data

1. Press STOP.
2. Set the STORAGE TEST switch to STEP.
3. Turn the CE MODE SELECTOR switch to ALTER SAR.
4. Set the ADDRESS/DATA switches on the console to the address of the storage position holding data to be altered. For Models 12C and 15, set both the ADDRESS/DATA switches on the console and the CE address switch(es) on the CE panel to the address of the storage position holding data to be altered.
5. Turn display roller to setting 3.
6. Press START.
7. Turn the CE MODE SELECTOR to DISPLAY STOR.
8. Press START. The byte stored will be displayed in the Q-register display lights.

To resume normal operation it is necessary to set the storage address register to the address of the instruction with which you wish to begin processing.

Displaying Local Storage Registers except ATT and PMR

1. Press STOP.
2. Turn the register display roller switch to LSR HI/LSR LO (roller position 2).
3. Turn the LSR display selector switch to the desired LSR.

Displaying ATT or PMR Data (Models 12C and 15)

1. Press STOP.
2. Turn the register display roller switch to position 6 to display the data in the ATT register, or to position 7 to display the data in the PMR.
3. Turn the CE mode selector switch to DISPLAY ATT/PMR.
4. Set the number of the register to be displayed into the leftmost two rotary switches on the console. (See Figure 4-20 for the ATT and PMR identification numbers.)
5. Press START.

Altering ATT or PMR Contents (Models 12C and 15)

1. Press STOP.
2. Turn the CE mode selector switch to ALTER ATT/PMR.
3. Set the number of the register to be altered into the leftmost two rotary switches on the console. (See Figure 4-20 for the ATT and PMR identification numbers.)
4. Set the data to be loaded into the register in the rightmost two rotary switches on the console.
5. Press START.

PROGRAM CHECK RECOVERY PROCEDURES–MODEL 15 ONLY

Program check interrupts allow a program error that occurs in one partition to be handled without always stopping the entire processing of the other partition. Processor checks cause the system to come to an immediate stop, and cause all I/O data transfer to stop immediately. Therefore, processor checks stop programs in the processing unit. Figure 4-21 defines the checks and the suggested action.

UNIT CHECK CONDITION

The program tests check indicators in the I/O units to detect unit checks. Whenever the program detects a unit check, it initiates a programmed halt with a halt identifier displayed on the message display unit; this identifier should be keyed to an operator restart/recovery procedure listing.

Name of Check	Cause	Suggested Action
Invalid address (see note)	Storage address register is addressing a location outside the available storage.	Log the error and provide operator message to cancel the job or to continue with the job. If the job is to be continued, provide operator instructions. If program must be corrected, correct the program.
Invalid operation (see note)	The operation code in the op register (from the instruction being processed) is invalid.	
Invalid Q-byte (see note)	No I/O device recognized the I/O instruction because: – The addressed device is not attached to the system. – The instruction N-code is invalid. If this check and privileged check are both indicated, a privileged instruction issued in non-privileged mode was detected during the I-Q cycle.	
Storage protect (see note)	The program attempted to access or write into protected storage.	
Privileged (see note)	CPU detected a privileged instruction while in nonprivileged state.	
Parity	CPU detected incorrect parity.	Operator must perform an IPL operation. Point of restart is a program/operator function. If program must be corrected, correct it before restarting job.

Note: If this check is encountered in interrupt level 7, or is encountered in any other level while interrupt level 7 is disabled, the check causes a processor stop. Also, if an invalid address is encountered during an I/O cycle, the check causes a processor stop. Suggested action for a processor stop is the same action suggested for a parity check.

Figure 4-21. Checks and Suggested Procedures

IBM 1442 Card Read Punch

An IBM 1442 Card Read Punch Model 6 or 7 can be attached to an IBM System/3 to provide 80-column card reading and punching (Figure 5-1).

The following operations can be performed on the 1442:

- Feed
- Read column binary
- Read translate
- Punch and feed
- Punch and no feed

Each of these operations is initiated by a start I/O instruction. The operation is specified by the Q-byte (byte 2) and the R-byte (byte 3) of the instruction, called a control code.

1442 NOT-READY-TO-READY INTERRUPT—MODEL 15 ONLY

If interrupt level 6 is enabled, the 1442 sends an interrupt request to the system whenever the 1442 goes from a not-ready state to a ready state.

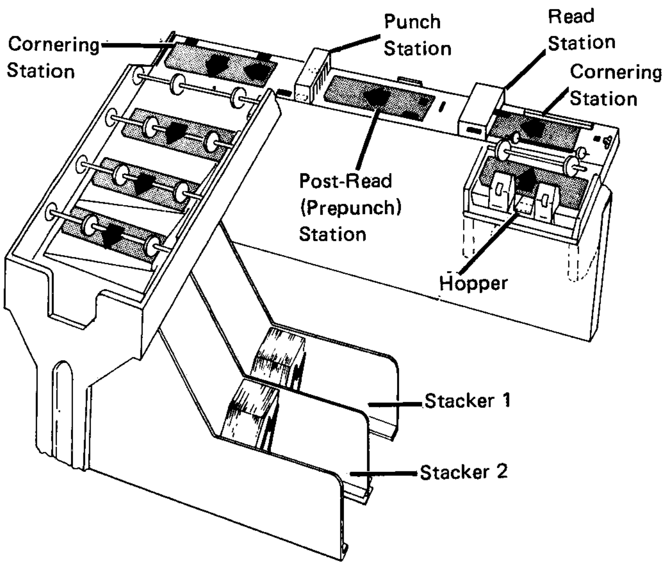


Figure 5-1. 1442 Card Path

1442 OPERATOR PANEL

Figure 5-2 shows the operator panel.

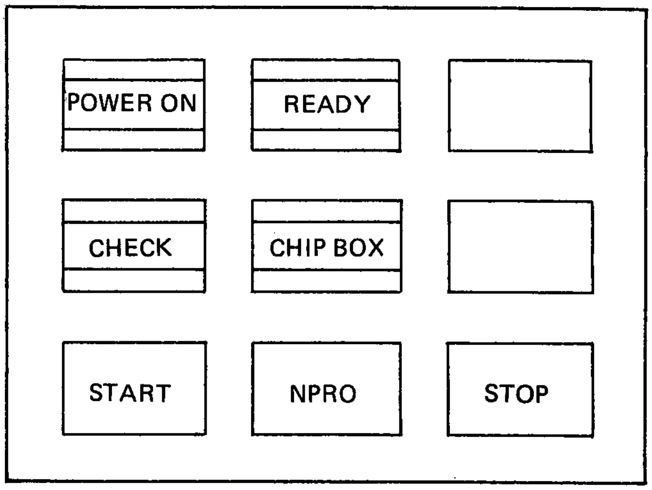


Figure 5-2. 1442 Operator Panel

POWER ON Light

This light indicates that system power is on.

READY Light

This light indicates that the 1442 is ready for processing. Pressing START, when all of the following conditions apply, turns READY on:

1. System power is on.
2. Cards are in the hopper.
3. Stacker is not full.
4. CHECK and CHIP BOX are off.
5. The 1442 covers are closed.

HOPR	FEED CLU	READ REG
READ STA		PUNCH
PUNCH STA		OVER RUN
TRANS		

Figure 5-3. 1442 Error Indicators

CHECK Light

This light turns on when any of the following error indicators turn on (Figure 5-3):

1. HOPR indicates that a card did not feed from the hopper when the 1442 took a feed cycle with cards in the hopper.
2. FEED CLU indicates that cards in the card path advanced one position because of an unrequested feed cycle.
3. READ REG indicates a read error.
4. READ STA indicates a card jam at the read station.
5. PUNCH indicates a punch error.
6. PUNCH STA indicates a card jam at the punch station.
7. OVERRUN indicates that data was lost because the processing unit was unable to accept data from the 1442 or send data to the 1442 fast enough.
8. TRANS indicates a card jam in the stacker area.

CHIP BOX Light

This light indicates that the chip box is full or out of place.

START Key

This key places the 1442 in ready status if the following conditions apply:

1. System power is on.
2. Cards are in the hopper.
3. Stacker is not full.
4. CHECK and CHIP BOX are off.
5. The 1442 covers are closed.

The start key is also used to return the 1442 to a ready status after the 1442 STOP key has been pressed.

NPRO Key

Pressing this key while the hopper is empty clears all cards from the card feed path. The first card that enters the stacker after NPRO is read but not punched. The second card that enters the stacker is not read or punched.

This key does not function if there are cards in the hopper.

STOP Key

This key stops the 1442 after the operation in process is completed.

1442 OPERATIONS

Read Operations

A load-I/O instruction must be executed before each start-I/O instruction that specifies card reading. This load-I/O instruction must load the address of the high-order byte of the read data field into the 1442 data address register. To meet performance specifications, the address for a normal read must be on a 128-byte boundary; the address for a read column binary must be on a 256-byte boundary.

The feed/read functions of start I/O instructions move cards from the hopper through the read station. If read is specified, the data contained in all 80 columns of the card is transferred to a storage field (1442 data field) specified by a load I/O instruction. The data read is checked to ensure that it is read correctly. An error in reading causes a read check.

The card feeding and reading rate is determined by the operations being performed. The rated reading speeds (300 cards per minute for Model 6 and 400 cards per minute for Model 7) are for read operations only. If punching is performed at the same time, the reading rate is reduced to the rate at which punching is performed. To maintain the rated reading rate, successive start I/O instructions specifying reading must be issued within 40 milliseconds (Model 6) or 30 milliseconds (Model 7) after the preceding card is read.

To test for a busy condition, use a test-I/O-and-branch instruction.

1442 IPL Read Operation—Model 15 Only

Pressing the PROGRAM LOAD key on the processing unit when the PROGRAM LOAD SELECTOR switch is set at ALTERNATE causes (1) the processing unit to load 0000 into the 1442 read data address, (2) the 1442 to read a card into storage, starting at address 0000, and (3) the processing unit to execute the instruction at position 0000. The IPL read operation occurs without execution of a 1442 start I/O instruction; otherwise, the read operation performed is similar to the usual 1442 start I/O read operation.

Punch Operations

A load-I/O instruction must be executed before each start-I/O instruction that specifies a punch operation. This load-I/O instruction places the address of the high-order byte of the punch data field in the 1442 data address register. Column 1 of the card is punched with the data in storage at this address; column 2 is punched with the data in storage at the next higher address. The punch data fields must be on 128-byte boundaries.

In addition to loading the 1442 data address register, a load-I/O instruction must be issued to load the length count register with 128 minus the number of columns to be punched.

Start-I/O instructions that specify punching move a card from the read station to the punch station. If a punch and feed command is issued, the card is punched and ejected into one of the stackers. If a punch with no feed command is issued, the card is punched but is not ejected.

As the cards pass through the punch station, data from storage is recorded in the cards in the form of punched holes. The punching is checked to ensure that the data is punched correctly. An error causes a punch check.

Card punching is performed at a rate of 80 columns per second (Model 6) or 160 columns per second (Model 7). For example:

Last Column Punched	Cards/Minute Model 6	Cards/Minute Model 7
1	260	355
40	84	145
80	49	91

To maintain the best card throughput, confine punching to the beginning card columns.

Combined Operations

Through proper sequencing of start I/O instructions, a card can be read and punched during one pass through the 1442.

Stacker Selection

Stacker selection is done by including the stacker select information in the start I/O instruction control code. Stacker selection is performed on the card that is in the punch station when the start I/O instruction is executed. If no stacker select information is given, the cards are automatically routed to stacker 1.

Op End Conditions—Model 15 Only

The 1442 op end occurs when one of the following happens:

- The 1442 goes from busy to not-busy at the end of a feed instruction.
- The 1442 goes from busy to not-busy at the end of a read instruction.
- The 1442 goes from busy to not-busy at the end of a punch instruction. Busy drops after the last column is punched on a punch with no feed instruction.

Note: If a feed check exists, the attachment sets the no-op status bit in response to any SIO issued and presents interrupt request at the end of the SIO instruction (during the IR cycle).

Any valid 1442 SIO can be used to enable, disable, or reset the interrupt request. In addition, an SIO instruction with N-code of 101 is available for interrupt control only. This instruction is accepted during busy, not ready, and unit check conditions.

1442 START I/O (SIO)

Op Code (hex)	Q-Byte (binary)	R-Byte (binary)
Byte 1	Byte 2	Byte 3
F3	0101 0 xxx	000x xxxx

DA M N Control Code

Control Code:

Bits 0123 4567	Function Specified	
000x x000[1]	Select stacker 1 for card at the punch station	
000x x001[1]	Select stacker 2 for card at the punch station	
	Model 15	*Model 12*
000x 100x[2]	Enable interrupt	Enable op-end indicator
0000 000x[2]	Disable interrupt	Disable op-end indicator
0001 x00x[2]	Reset interrupt	Reset op-end indicator

N:

N-Code	Function Specified
000	Feed
001	Read only
010	Punch and feed
011	Read column binary
100	Punch with no feed
101	*Model 10:* Invalid N-code; causes processor check
	Model 12: Perform op-end indicator control
	Model 15: Perform interrupt control

Any N-code not shown is invalid and causes:
- Program check if interrupt level 7 is enabled on Model 15
- Processor check if interrupt level 7 is not enabled on Model 15
- Processor check on Models 10 and 12

M: Not used; should be 0.

DA: Hex 5 specifies the 1442 as the device to be controlled.

F3 specifies a start I/O operation. F as the first hex character in the op code identifies a command-type instruction (that is, an instruction without operand addressing).

[1] This code controls stacker selection when the N-code specifies a card function (N≠101).
[2] This code does not control stacker selection when the N-code specifies an interrupt control function (N=101).

Operation

The 1442 performs the function specified by the N-code and control code (R-byte).

Program Notes

- If a 1442 check prevents execution of the SIO, the processing unit aborts the instruction. Meanwhile, the attachment sets the 1442 no-op status bit. On a Model 12, the attachment also sets the op-end indicator if the indicator is enabled. On a Model 15, the attachment also sets the op-end interrupt request if interrupts are enabled.

  If a 1442 check does not prevent execution of the instruction, the 1442 executes the instruction and the attachment resets the check. Feed checks and not-ready conditions cause no-op functions.

- The 1442 on Model 15 always accepts an SIO that specifies an interrupt control function. The 1442 on Model 12 always accepts an SIO that specifies op-end indicator control.

1442 TEST I/O AND BRANCH (TIO)

Op Code (hex)	Q-Byte (binary)	Operand Address	
Byte 1	Byte 2	Byte 3	Byte 4
C1	0101 0 xxx	Operand 1 address	
D1	0101 0 xxx	Op 1 disp from XR1	
E1	0101 0 xxx	Op 1 disp from XR2	

DA M N

N-Code	Condition Tested
000	Not ready/unit check. This condition indicates one of the following has occurred: Feed check (not ready) Read check (unit check) Punch check (unit check) No-op (unit check or not ready) I/O attention (not ready)
010	Busy. This condition indicates that the 1442 is feeding, reading, or punching a card.
101	*Models 8 and 10:* Invalid N-code; causes processor check *Model 12:* Op-end indicator on *Model 15:* Interrupt pending

Any N-code not shown is invalid and causes:
- Program check if interrupt level 7 is enabled on Model 15
- Processor check if interrupt level 7 is not enabled on Model 15
- Processor check on Models 10 and 12

M: Not used; must be 0.

DA: Hex 5 specifies the 1442 as the tested device.

C1, D1, or E1 specifies a test I/O and branch operation. The first hex character in the op code specifies the type of operand addressing for the instruction.

Operation

The processing unit tests the 1442 for the conditions specified by the N-code. If any one of the tested conditions exists, the program branches to the address in the operand portion of this instruction. If no tested condition exists, the program proceeds with the next sequential instruction.

Resulting Condition Register Setting

This instruction does not affect the condition register.

1442 ADVANCE PROGRAM LEVEL (APL)

Op Code (hex)	Q-Byte (binary)	R-Byte (binary)
Byte 1	Byte 2	Byte 3
F1	0101 0 xxx	0000 0000

DA M N — R-byte is not used for an APL instruction

N-Code: Condition Tested

N-Code	Condition Tested
000	Not ready or unit check. This condition indicates one of the following occurred: Feed check (not ready condition) Read check (unit check condition) Punch check (unit check or not ready condition) I/O attention (not ready condition)
010	Busy. This condition indicates that the 1442 is feeding, reading, or punching a card.
101	*Models 8 and 10:* Invalid N-code; causes processor check *Model 12:* Op-end indicator on *Model 15:* Interrupt pending

Any N-code not shown is invalid and causes:
- Program check if interrupt level 7 is enabled on Model 15
- Processor check if interrupt level 7 is not enabled on Model 15
- Processor check on Models 10 and 12

M: Not used; must be 0.

DA: Hex 5 specifies the 1442 as the tested device.

F1 specifies an APL operation. F as the first digit in the op code identifies a command-type instruction (that is, an instruction without operand addressing).

Operation

This instruction tests for the conditions specified in the Q-byte.

- Condition present:
  - Systems with Dual Program Feature installed and enabled, activate the inactive program level.
  - Systems without Dual Program Feature installed or with Dual Program Feature installed but not enabled, loop on the advance-program-level instruction until the condition no longer exists.

- Condition not present: Systems with or without Dual Program Feature access the next sequential instruction in the active program level.

Program Note

For additional information concerning the advance program level instruction, see Chapter 2.

1442 LOAD I/O (LIO)

Op Code (hex)	Q-Byte (binary)	Operand Address	
Byte 1	Byte 2	Byte 3	Byte 4
31	0101 0 xxx	Operand 1 address	
71	0101 0 xxx	Op 1 disp from XR1	
B1	0101 0 xxx	Op 1 disp from XR2	

DA M N

N-Code	To Be Loaded
000	Length count register
100	1442 data address register

Any N-code now shown is invalid and causes:

- Program check if interrupt level 7 is enabled on Model 15
- Processor check if interrupt level 7 is not enabled on Model 15
- Processor check on Models 10 and 12

M: Not used; should be 0.

DA: Hex 5 specifies the 1442 as the addressed device.

31, 71, or B1 specifies a load I/O operation. The first hex character in the op code specifies the type of operand addressing to be used for the instruction.

Operation

The processing unit loads the 2 bytes of data contained in the operand into the register specified by the N-code. The operand is addressed by its low-order (higher numbered) storage position.

Program Note

If the selected register is busy, the program loops on the load I/O instruction until the register is not busy.

1442 SENSE I/O (SNS)

Op Code (hex)	Q-Byte (binary)	Operand Address	
Byte 1	Byte 2	Byte 3	Byte 4
30	0101 0 xxx	Operand 1 address	
70	0101 0 xxx	Op 1 disp from XR1	
B0	0101 0 xxx	Op 1 disp from XR2	

DA M N

N-Code Sensed Unit

001 CE diagnostic indicators
010 CE diagnostic indicators
011 Status indicators
100 1442 data address register

Any N-code not shown is invalid and causes:
- Program check if interrupt level 7 is enabled on Model 15
- Processor check if interrupt level 7 is not enabled on Model 15
- Processor check on Models 10 and 12

M: Not used; should be 0.

DA: Hex 5 specifies the 1442 as the device being sensed.

30, 70, or B0 specify a sense I/O operation. The first hex character is the op code specifies the type of operand addressing for the instruction.

Operation

The 1442 transfers 2 bytes of data from the unit specified by the N-code to the main storage field specified by the operand address. The first byte transferred enters the effective address (the operand address), the second byte enters the effective address minus 1.

Byte	Bit	Name	Indicates	Reset By
1	0	Read check	Data read from the card may be invalid. (This bit sets the condition that can be tested by a TIO instruction, and turns on the CHECK and READ REG lights.)	Next SIO accepted by 1442, system reset, NPRO operation, or check reset
1	1	Last card	The 1442 has stopped because of an empty hopper condition and the operator has pressed START without refilling the hopper (*example:* end-of-job routine). The bit turns on when the card passes the read station when START is pressed. With the bit on, there is a card ahead of the punch station. Only a feed, punch and feed, or punch only command should be issued while the last card bit is on. A punch only command does not move the card and the bit remains on. A command specifying a feed moves the card into the stacker and turns the bit off.	NPRO operation or a subsequent instruction specifying a feed operation
1	2	Punch check	The correct punches were not set, so the card was incorrectly punched. (This bit sets the condition that can be tested by the TIO instruction and turns on the 1442 PUNCH and CHECK lights.)	Next SIO accepted by 1442, system reset, NPRO operation, or check reset
1	3	Data overrun	Data was lost because the CPU was unable to accept data from the 1442 or send data to the 1442 fast enough. (Data overrun turns on the OVERRUN and CHECK lights.)	Next SIO accepted by 1442, system reset, NPRO operation, or check reset
1	4	Not ready	The 1442 requires operator intervention because the hopper is empty, the stacker is full, the chip box is full or not installed properly, a cover is open, or STOP was pressed. If the chip box caused the not-ready condition, the CHIP BOX light is on.	Correcting the condition that caused the not-ready state and pressing START
1	5	No-op	The program issued a command the 1442 accepted but was unable to execute because of a previous error. (This bit sets a condition that can be tested by the TIO instruction.)	Next sense instruction accepted by the 1442
1	6	Feed check	Improper card movement in the card path of the 1442. (This bit also sets a condition that can be tested by the TIO instruction, makes the 1442 not-ready, turns on the CHECK light, and lights the appropriate error indicator.)	Correcting the condition causing the check and pressing START
1	7	Read invalid	The 1442 detected multiple punching in rows 1 through 7 of a single column in the card being read. This can be caused by invalid punching or by cards being inserted in the hopper such that the 12-edge feeds first. (This bit turns on the READ REG and CHECK lights.)	Next SIO accepted by 1442, system reset, NPRO operation, or check reset

Figure 5-4 (Part 1 of 2). 1442 Status Bytes

Byte	Bit	Name	Indicates	Reset By
2	0	Not used		
2	1	Not used		
2	2	Not used		
2	3	Read station jam	The read station operated improperly or a card misfeed occurred at the read station. (The 1442 goes not-ready and the READ STA and CHECK lights turn on when the 1442 sets this bit.)	NPRO operation
2	4	Hopper misfeed	Card failed to feed properly from the hopper. (This bit makes the 1442 not-ready and turns on the HOPR and CHECK lights.)	NPRO operation
2	5	Extra feed cycle	The 1442 took an unrequested feed cycle. (The 1442 goes not-ready and the FEED CLU and CHECK lights turn on when the 1442 sets this bit.)	NPRO operation
2	6	Punch station jam	A card misfeed occurred at the punch station. (This bit makes the 1442 not-ready and turns on the PUNCH STA and CHECK lights.)	NPRO operation
2	7	Transport jam	A card misfeed occurred in the stacker transport area. (This condition makes the 1442 not-ready and turns on the TRANS and CHECK lights.)	NPRO operation

Figure 5-4 (Part 2 of 2). 1442 Status Bytes

IBM 2501 Card Reader

The IBM 2501 Card Reader Model A1 or A2 can be attached to IBM System/3 as an 80-column punched card input device. Model A1 reads cards at a maximum rate of 600 cards per minute. Model A2 reads cards at a maximum rate of 1000 cards per minute. Each model has a hopper capacity of 1200 cards and a stacker capacity of 1300 cards.

The processing unit program controls card reading. The 2501 reads cards column-by-column (serially) beginning in column 1, reading each column twice and comparing the two readings to check reading accuracy. As a further check on reading accuracy, the 2501 attachment detects off-punched cards, mispositioned cards, and hardware failures.

2501 NOT-READY-TO-READY INTERRUPT—MODEL 15 ONLY

If interrupt level 6 is enabled, the 2501 sends an interrupt request to the system whenever the 2501 goes from a not-ready state to a ready state.

2501 CARD PATH

As shown in Figure 5-5, the 2501 card path consists of the hopper, the preread station, the read station, the stacker, and the transport between the hopper and stacker. Two cycles are required to move each card from the hopper to the stacker. During the first cycle, the card moves from the bottom of the stack of cards in the hopper into the preread station. During the second card feed cycle, the card moves from the preread station, through the read station, and into the stacker.

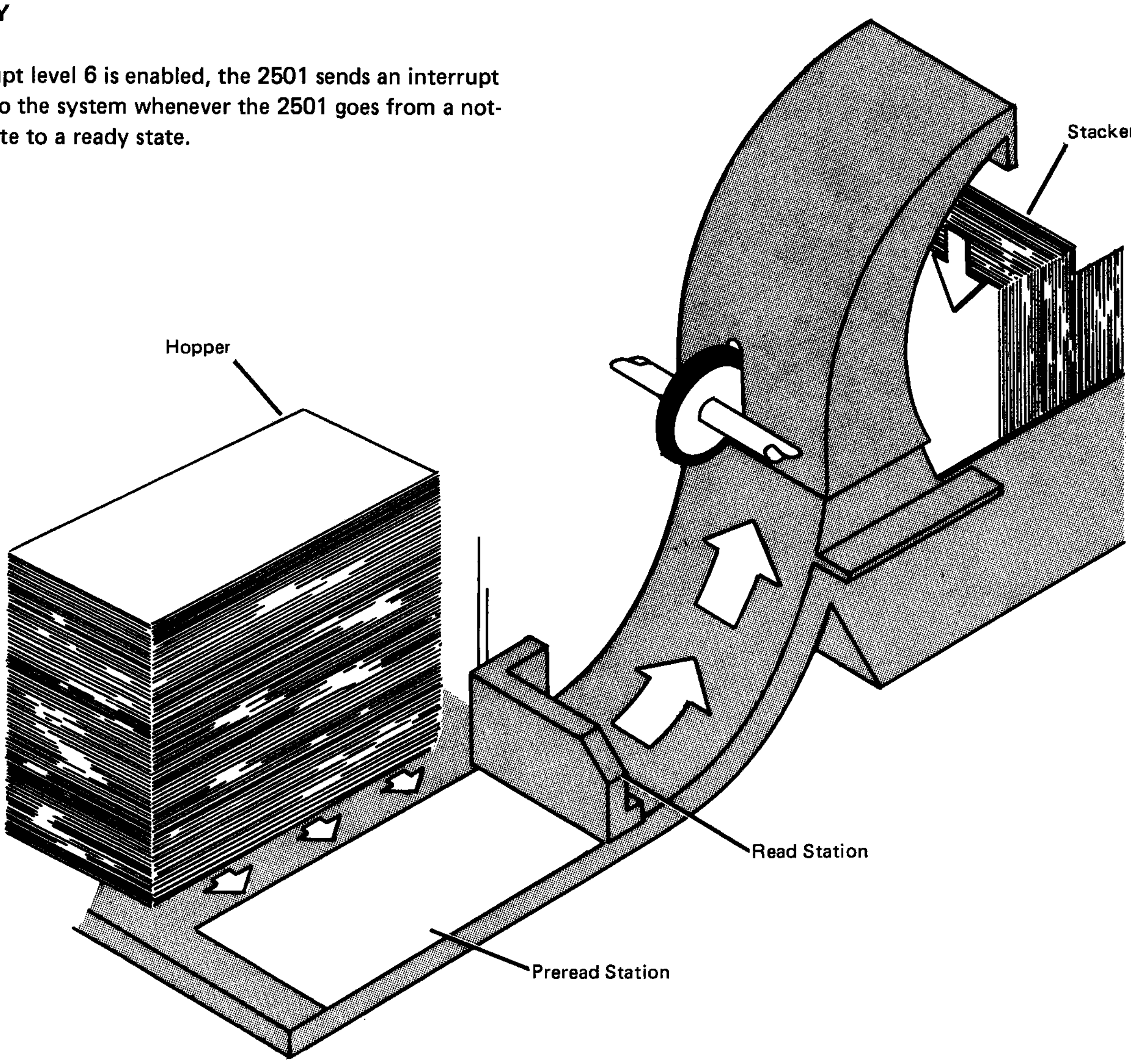


Figure 5-5. 2501 Card Path

2501 READ STATION

Figure 5-6 shows the 2501 read station and explains the reading principle.

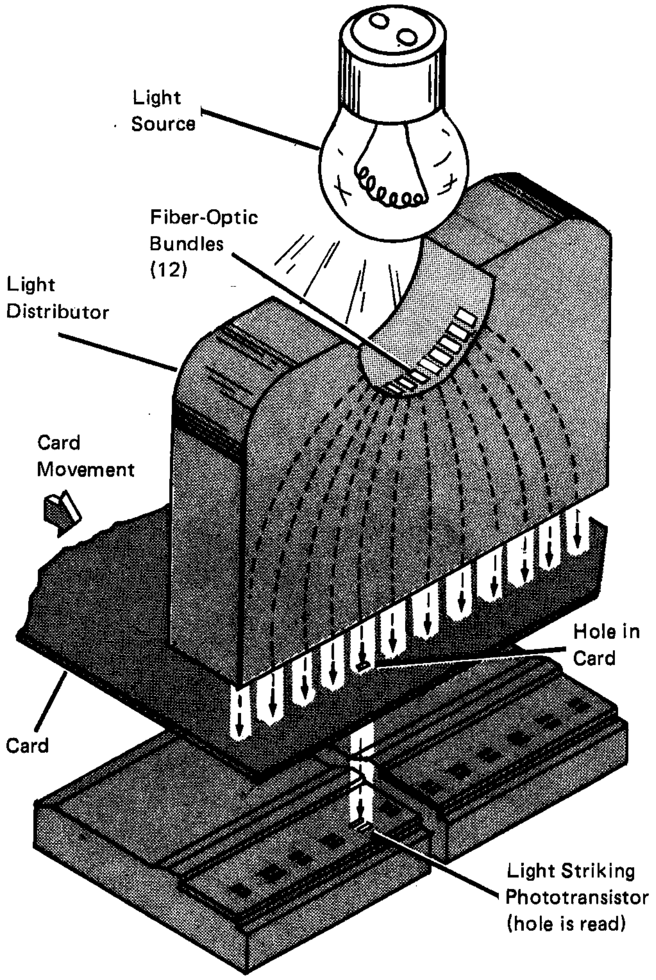


Figure 5-6. 2501 Read Station

2501 KEYS

Figure 5-7 shows the 2501 operating keys.

2501 START Key

If the preread station is empty and there is at least one card in the hopper, pressing START initiates a run-in cycle that moves the bottom card from the hopper into the preread station and makes the 2501 ready.

If there is a card at the preread station and the 2501 is not ready, pressing START makes the 2501 ready.

Exception: START is not functional if any of the following conditions apply:

Feed check condition exists
Stacker is full
2501 cover is open
STOP is also being pressed

2501 STOP Key

Pressing STOP makes the 2501 not-ready. If the 2501 is performing a functional cycle when you press STOP, ready drops at the end of the cycle; otherwise, ready drops immediately. Pressing STOP has no effect while the 2501 is not ready.

2501 NPRO (Nonprocess Runout) Key

Pressing NPRO while there is no card in the hopper moves the card from the preread station into the stacker without reading it. This clears the card path of all cards.

Note: If cards are mispositioned or jammed in the card path, remove them manually before pressing NPRO to clear the feed check condition.

2501 LIGHTS

Figure 5-7 shows the operator panel, which contains the 2501 lights.

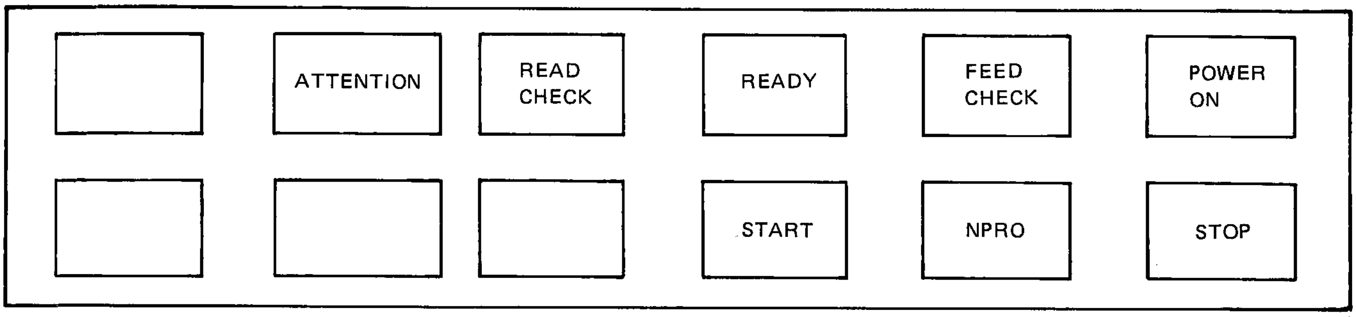


Figure 5-7. 2501 Operator Panel

2501 POWER ON Light

This light indicates that power is being supplied to the 2501.

2501 READY Light

This light indicates that the 2501 is ready and can process cards. The light therefore indicates:

- Card in the preread station
- Stacker not full
- No feed check conditions
- 2501 covers closed
- STOP not pressed since the 2501 was last made ready
- Hopper not empty, or 2501 in last card condition (that is, the hopper is empty, causing the 2501 to go not ready, then the operator pressed START to make the 2501 ready again to accept an SIO read command to read the card in the preread station and stack that card)

2501 FEED CHECK Light

This light indicates one of the following:

- Card misfeed in the card path
- Equipment malfunction
- Power-on sequence

Correct the error condition (if any) and press NPRO to turn FEED CHECK off. If repeated feed checks occur, notify your customer engineer.

2501 READ CHECK Light

This light indicates that the attachment detected one of the following conditions:

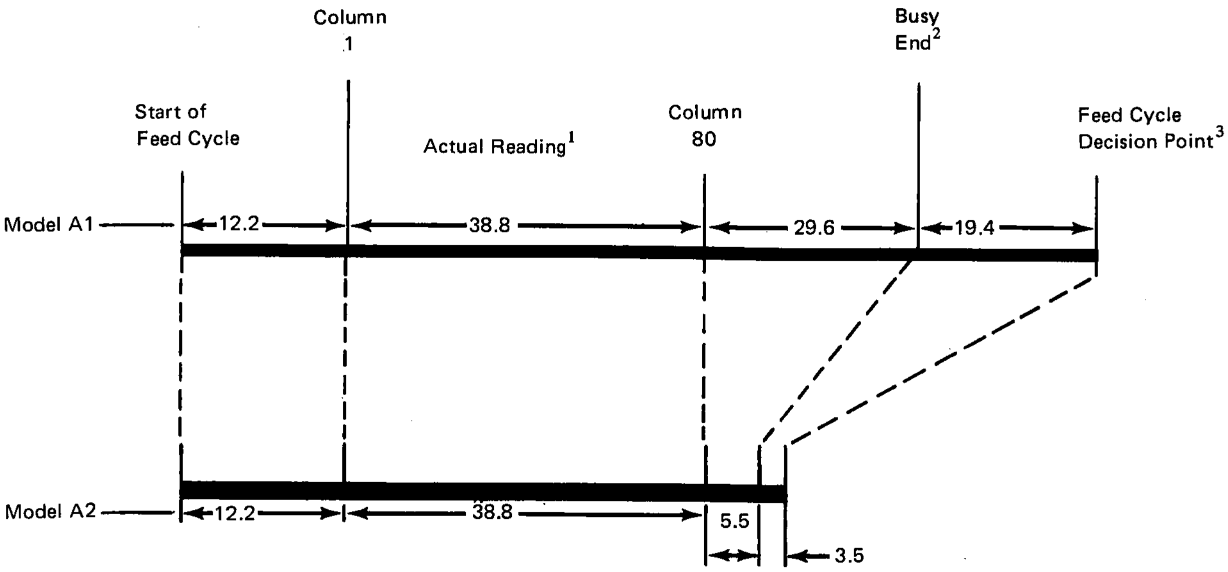
- Read emitter failure (machine failure)
- Fiber optics failure (machine failure)
- Read overrun (The processing unit did not grant a cycle steal soon enough to transfer data from one column before the 2501 started to read the next column. This resulted in the loss of one column of data.)
- Card punched off-registration or placed in the hopper with the 9-edge of the card next to the 12-edge side of the hopper (the attachment sends a hex 00 to the processing unit instead of the byte in error).
- Invalid card code (while reading in translate mode, the attachment detected more than one punch in rows 1 through 7 of a single card column. This could also be caused by the card being placed in the hopper with the 9-edge toward the 12-edge side. The attachment sends a hex 00 to storage in place of the byte in error).
- Translate check (machine error)

2501 ATTENTION Light

This light indicates that operator intervention is required at the 2501 to do one of the following:

- Close a cover
- Empty the stacker

Correcting the condition causing the light and pressing either START or NPRO turns ATTENTION off.



[1] Reading stops on length count register overflow, which occurs after last column specified to be read has been read.
[2] Busy end sets an op-end interrupt request (if enabled) on Model 15 and operation end indicator (if enabled) on Model 12.
[3] SIO read instruction should be issued between busy end and feed cycle decision point for maximum card read rate.

Note: Timings are shown in milliseconds.

Figure 5-8. 2501 Read/Feed Timings

2501 READ OPERATION

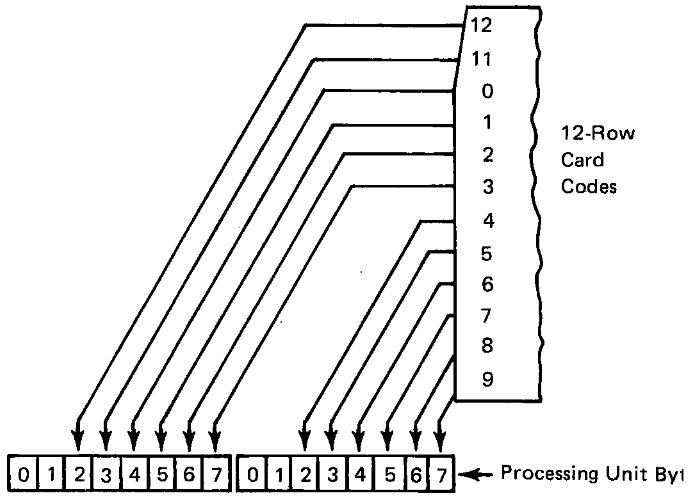
A complete read operation requires the program to load the storage address of the read field into the data address register and to indicate the number of columns to be read from the card by loading a value into the length count register. (This requires two LIO instructions.)

After the data length and input area have been defined, the program can issue an SIO read instruction. This instruction moves a card from the preread station, past the read head and into the stacker. The 2501 reads the specified number of columns from the card as the card passes the read station. The 2501 feeds a new card from the hopper to the preread station during the same card cycle.

When the 2501 goes not busy, near the end of the feed cycle (Figure 5-8), interrupt request occurs on level 5 if interrupts were enabled on Model 15 and the operation end indicator turns on (if enabled) on Model 12.

Data can be transferred from cards into main storage in either of two modes: card image mode (Figure 5-9) or translate mode (Figure 5-10). In card image mode, 2 bytes are transferred without translation, into main storage for each card column read. In translate mode, 1 byte is transferred to main storage for each card column read. In this mode, punching read from the 12 punch positions in each column is translated into an 8-bit EBCDIC byte.

Attaining the maximum reading rate for each model of the 2501 is a programming function, and can be achieved by always issuing the SIO read instruction prior to the feed cycle decision point (Figure 5-8). If the instruction is issued after the feed cycle decision point, the 2501 awaits the next decision point before starting the read operation.



Note: Attachment puts zeros in bytes 0 and 1.

Figure 5-9. 2501 Card Image Read Mode

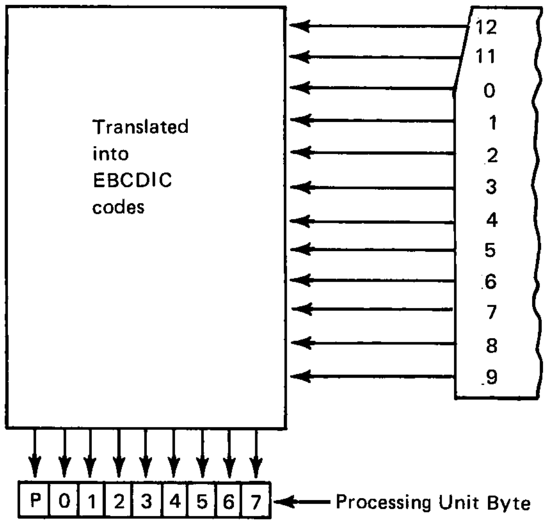


Figure 5-10. 2501 Translate Read Mode

2501 START I/O (SIO)

Op Code (hex) Byte 1	Q-Byte (binary) Byte 2	R-Byte (binary) Byte 3
F3	0011 1 xxx	xxxx xxxx

DA M N — Control Code

Bits

0123 4567[1]	Control Function Specified	
	Model 15	*Model 12*
00x0 1000	Enable interrupt	Enable op-end indicator
00x0 0000	Disable interrupt	Disable op-end indicator
0010 x000	Reset busy end (op-end) interrupt	Reset op-end indicator

N-Code	Function
000	*Model 12:* Control op-end indicator *Model 15:* Control interrupts
001	Read in translate mode (interrupt control is optional)
011	Read in card image mode (interrupt control is optional)

Any N-code not shown is invalid and causes:
- Program check if interrupt level 7 is enabled on Model 15
- Processor check if interrupt level 7 is not enabled on Model 15
- Processor check on Model 12

00111 specifies the 2501 as the device to be controlled.

F3 specifies a start I/O operation. F as the first hex character of the op code designates the instruction as a command-type instruction (that is, with no operand addressing).

[1] Bits 0, 1, 5, 6 and 7 are not used; they should be set to 0.

Operation

The 2501 performs the operation specified by the N-code. If the operation is a read operation, the 2501 moves a card from the preread station, past the read head, and into the stacker, and feeds a card from the hopper into the preread station. As the first card moves past the read head, the 2501 reads the number of columns specified by the length count register into the main storage field specified by the data address register.

2501 TEST I/O AND BRANCH (TIO)

Op Code (hex)	Q-Byte (binary)	Operand Address	
Byte 1	Byte 2	Byte 3	Byte 4
C1	0011 1 xxx	Operand 1 address	
D1	0011 1 xxx	Op 1 disp from XR1	
E1	0011 1 xxx	Op 1 disp from XR2	

DA M N

N-Code	Condition Tested	
	Model 15	*Model 12*
000	Not ready/check	Not ready/check
001	Interrupt pending	Op-end indicator on
010	Busy	Busy

Any N-code not shown is invalid and causes:
- Program check if interrupt level 7 is enabled on Model 15
- Processor check if interrupt level 7 is not enabled on Model 15
- Processor check on Model 12

00111 specifies the 2501 as the device to be tested.

C1, D1, or E1 specifies a test-I/O-and-branch operation. The first hex digit in the op code signifies the type of operand addressing to be used for the instruction.

Operation

The processing unit tests the 2501 for the condition specified by the N-code. If the condition exists, the program branches to the address in the operand address portion of the instruction. If the condition does not exist, the program immediately accesses the next sequential instruction.

Program Notes

At the end of the TIO instruction, the IAR and ARR registers swap contents if a branch is to occur. That is, the instruction address register holds the address of the branch-to instruction and the address recall register holds the address of the next sequential instruction.

Resulting Condition Register Setting

This instruction does not affect the condition register setting.

2501 ADVANCE PROGRAM LEVEL (APL)

Op Code (hex) Byte 1	Q-Byte (binary) Byte 2	R-Byte (binary) Byte 3
F1	xxxx x xxx	0000 0000

DA M N

R-byte is not used in an APL instruction.

N-Code **Condition Tested**

N-Code	*Model 15*	*Model 12*
000	Not ready/check	Not ready/check
001	Interrupt pending	Op-end indicator on
010	Busy	Busy

Any N-code not listed is invalid and causes:
- Program check if interrupt level 7 is enabled on Model 15
- Processor check if interrupt level 7 is not enabled on Model 15
- Processor check on Model 12

00111 specifies the 2501 as the device to be tested.

F1 specifies an APL operation. The first digit of the op code (hex F) signifies that the instruction is a command-type instruction (that is, that no operand addressing is used).

Operation

This instruction tests for the conditions specified in the Q byte.

- Condition present:
  - Systems with Dual Program Feature installed and enabled, activate the inactive program level.
  - Systems without Dual Program Feature installed or with Dual Program Feature installed but not enabled, loop on the advance program level instruction until the condition no longer exists.

- Condition not present: Systems with or without Dual Program Feature access the next sequential instruction in the active program level.

Program Note

For additional information concerning the advance program level instruction, see Chapter 2.

Resulting Condition Register Setting

This instruction does not affect the condition register.

2501 LOAD I/O (LIO)

Op Code (hex)	Q-Byte (binary)	Operand Address	
Byte 1	Byte 2	Byte 3	Byte 4
31	0011 1 xxx	Operand 1 address	
71	0011 1 xxx	Op 1 disp from XR1	
81	0011 1 xxx	Op 1 disp from XR2	

DA M N

N-Code **Register Loaded**

000 Length count register. (Load 128 minus *n*, where *n* is the number of the last column to be read. The 2501 starts reading the first column of the card and continues reading until column *n* has been read.)

100 Data address register.

Any N-code not shown is invalid and causes:
- Program check if interrupt level 7 is enabled on Model 15
- Processor check if interrupt level 7 is not enabled on Model 15
- Processor check on Model 12

DA = 0011 and M = 1 specify the 2501 as the device to be loaded.

31, 71, or B1 specifies a load I/O operation. The first hex character in the op code specifies the type of operand addressing for the instruction.

Operation

The processing unit loads the 2 bytes of data contained in the operand into the register specified by the N-code. The operand is addressed by its low-order (higher numbered) storage position.

Program Note

If a length count of 0 is specified, the processing unit aborts the next SIO read command, sets the no-op status bit, and requests an op-end interrupt.

2501 SENSE I/O (SNS)

Op Code (hex)	Q-Byte (binary)	Operand Address	
Byte 1	Byte 2	Byte 3	Byte 4
30	0011 1 xxx	Operand 1 address	
70	0011 1 xxx	Op 1 disp from XR1	
B0	0011 1 xxx	Op 1 disp from XR2	

DA M N

N-Code	Sensed Unit
001	Status byte 4 and a meaningless byte[1]
010	Status byte 2 and 3[1]
011	Status bytes 0 and 1[1]
100	2501 data address register

Any N-code not shown is invalid and causes:
- Program check is interrupt level 7 is enabled on Model 15
- Processor check if interrupt level 7 is not enabled on Model 15
- Processor check on Model 12

DA = 0011 and M = 1 specify the 2501 as the device to be sensed.

30, 70, or B0 specifies a sense I/O operation. The first hex character in the op code specifies the type of operand addressing for the instruction.

---

[1]Even-numbered bytes (including byte 0) are stored at the storage position specified by the operand address. Odd-numbered bytes are stored at the operand address minus 1.

Operation

The processing unit stores the data specified by the N-code into the storage data field specified by the operand address. The operand is a 2-byte field and is addressed by its higher numbered position. Figure 5-11 defines the bits in the status bytes and describes their meanings.

Program Note

By examining the contents of the data address register, you can determine how many columns of card data were moved to the 2501 data field (the data address register always indicates the address of the last position filled plus 1).

Byte	Bit	Name	Indicates	Reset By
0	0	Preread station feed check	A card failed to feed from the preread station to the read station.	Correcting the condition that caused the check, then pressing NPRO
0	1	Read station feed check	A card misfeed occurred at the read station, a read cell was covered with dust or paper scraps, or a read cell failed electrically.	
0	2	Hopper feed check	A card failed to feed from the hopper to the preread station.	
0	3	Invalid card code	Multiple punches were read in a single card column by row sensors 1 through 7. This could be caused by a card having been placed in the hopper with its 9-edge toward the 12-edge side of the hopper. When the check occurs, the attachment sends hex 00 to main storage instead of the invalid character read, but does not halt the 2501.	System reset key, check reset key, or the next SIO accepted by the attachment (Although the attachment does not halt the 2501 for this type of check, the program should stop the reader and provide an operator procedure to reprocess the card causing the check.)
0	4	Unequal compare check	The card read had poor punch registration or the card causing the check was placed in the hopper with its 9-edge toward the 12-edge side of the hopper. When this check occurs, the attachment sends hex 00 to the processing unit instead of the character from the column being read when the check occurred. This check does not halt the 2501.	
0	5	Fiber optics check	The read unit provided a false indication that a punch position contained a punched hole, although there was no hole punched in that punch position.	
0	6	Read overrun	The processing unit did not grant a cycle steal in time to accept data from one column of the card before the 2501 presented data from the next column of the card. Therefore, 1 byte of data was lost.	
0	7	No read emitter check	Read emitter failure while a card was being read.	
1	0	Length count register overflow	The last column was read from the card.	Next SIO read or load length count register instruction accepted by 2501
1	1	Trailing edge	–	
1	2	Card reader ready	The 2501 is in a ready status. (This condition can also be tested by a TIO instruction.)	

Figure 5-11 (Part 1 of 2). 2501 Card Reader Status Bytes

Byte	Bit	Name	Indicates	Reset By
1	3	No-op	The program issued a command that the 2501 is not capable of executing, so the 2501 attachment accepts the command but does not execute it.	System reset, check reset, or the SNS instruction that senses this condition
1	4	Read complete	The read operation has been completed. (This indication is essentially the same as the length count overflow indication; however, loading the LCR does not reset the read complete bit.)	Next SIO instruction accepted by the 2501
1	5	Translate check	The attachment detected a parity error in translated data while in translate mode.	System reset, check reset, or the next SIO read instruction accepted by the attachment (Although the attachment does not halt the 2501 for this type of check, the program should stop the reader and provide an operator procedure to process the card being read when the check occurred. If the error persists, call your customer engineer.)
1	6	Not used; will be 1	Has no meaning.	
1	7			

Figure 5-11 (Part 2 of 2). 2501 Card Reader Status Bytes

IBM 2560 Multi-Function Card Machine

The IBM 2560 Multi-Function Card Machine (MFCM) is available in two models, the Model A1 and the Model A2, which read, punch, and collate 80-column cards under control of the system. Model A1 can be equipped with a special feature that permits the 2560 to print data on the cards.

Both models have two card hoppers, a light sensing read station that reads both primary and secondary cards, and a punch station common to the primary and secondary feeds. If the print special feature is installed, the A1 has a print station. The 2560 Model A1 has five radial stackers; the A2 has four. Cards can be directed into any of the stackers under program control. Unselected secondary cards enter stacker 5 of the Model A1 and stacker 4 of the Model A2; unselected primary cards enter stacker 1 for both models.

Multifunction capability permits collating, gangpunching, reproducing, summary punching, calculating, printing, and classifying of cards in one pass. Thus, the multifunction concepts brings to card processing a flexibility approaching that of magnetic tape drives and random access storage.

Both models of the 2560 can read and punch any of the 256 characters of the extended binary coded decimal interchange code (EBCDIC). Model A1 reads at a rate of 500 cards per minute and punches at a rate of 160 card columns per second; Model A2 reads at a rate of 310 cards per minute and punches at a rate of 120 card columns per second. If the Model A1 is equipped with the print special feature, printing occurs at a rate of 140 characters per second.

Separate primary and secondary hoppers, each with a 1200-card capacity, feed in cards parallel, 9-edge first, face down. Once out of the hopper, cards move serially in both card paths (Figure 5-12).

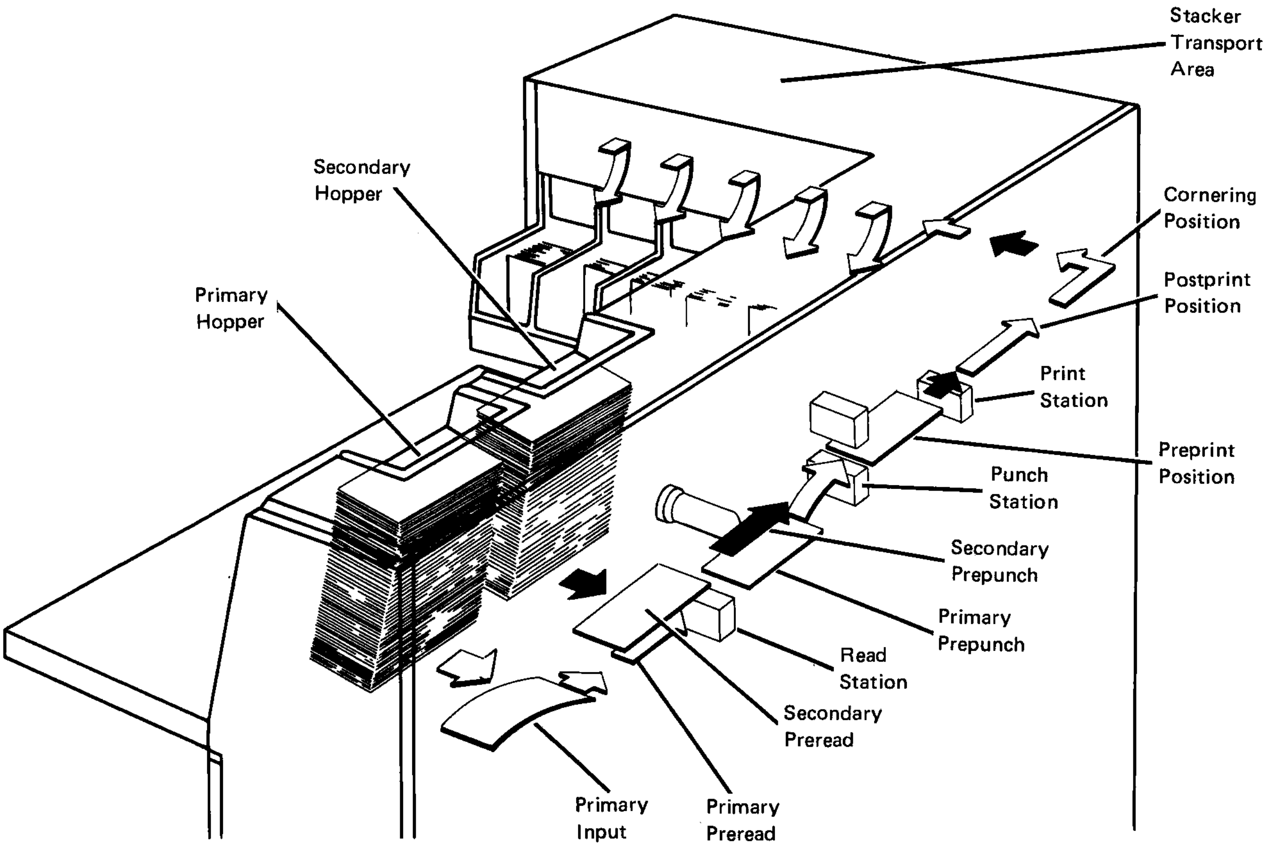


Figure 5-12. 2560 Card Paths

The primary and secondary card paths are separate through the preread and prepunch and share a common path through the read, punch and print stations. After leaving the print station, cards move serially to the cornering station then parallel to the selected stacker.

The processing unit stored program controls the movement of cards in the separate and merged paths, as well as to the radial stackers, which allow the selection and interfiling of cards from either feed.

2560 FEEDS AND TRANSPORT

2560 Primary Feed

The primary card hopper feeds cards in paralled from the bottom of the stack into the primary input station. Two card-feed cycles are required to place the first card into the primary preread station.

2560 Secondary Feed

The secondary card hopper feeds cards in parallel from the bottom of the stack directly into the secondary preread station. Only one card-feed cycle is required.

2560 Initial Read-In

When the system is turned on and power comes up in the 2560, a feed check occurs to ensure that no cards are left in the card path. To start operation, the operator presses the nonprocess runout (NPRO) key to clear the feed check, loads the hoppers with cards, and presses START. This makes ready the 2560 and completes the initial setup.

Pressing START causes one card to feed from the secondary hopper to the secondary preread station, and two cards to feed from the primary hopper. The first card to feed from the primary hopper is positioned at the primary preread station, and the second card is positioned at the primary input station.

The cards move serially as they are transported from the primary input and secondary preread stations.

2560 Read Station

The read station contains a 12-position light-cell assembly. Cards from either the primary feed or the secondary feed are read column-by-column, and the data is stored in the main storage.

When instructed by the processing unit program, the 2560 moves a card from the primary preread station through the read station to the primary prepunch station. A similar program instruction moves a card from the secondary preread station through the read station to the secondary prepunch station. All cards in the designated card path advance one station.

Each card column is read twice; the two readings are compared to check reading accuracy.

2560 Punch Station

As directed by the stored program, the 2560 moves a card from either the primary prepunch station or secondary prepunch station to the punch station, where the card is registered. The primary and secondary card paths converge into a single card path at the punch station. The card is punched serially, column-by-column.

If there is no card at the prepunch station when an SIO punch instruction occurs in the program, the instruction is no-oped.

The punching of each hole generates a signal that is automatically compared with the corresponding data from main storage to check punching accuracy.

2560 Card Stackers

As shown in Figure 5-12, a card enters one of the radial stackers after it has been processed by the 2560. The stackers are numbered from 1 to 5 on the Model A1 and from 1 to 4 on the Model A2. Cards are stacked in the sequence in which they are fed. Each stacker holds about 1300 cards and is equipped with a full-stacker switch that prevents over filling of the stacker.

2560 SPECIAL FEATURES

2560 Card Print

With the Card Print special feature installed, a card ejected from the punch station registers for printing. A stored program instruction causes the card to be printed.

If there is no card at the preprint station when an SIO print instruction occurs in the program, the instruction is no-oped.

A character set consisting of 10 numeric, 26 alphabetic, 26 special characters, and a blank character can be printed as sent from the processing unit (Figure 5-13).

Byte Positions 0 34 7	Character Printed	Card Code	Byte Positions 0 34 7	Character Printed	Card Code
01000000	blank				
01001010	¢	T28	11000110	F	T6
01001011	.	T38	11000111	G	T7
01001100	<	T48	11001000	H	T8
01001101	(	T58	11001001	I	T9
01001110	+	T68			
01001111	|	T78	11010001	J	E1
			11010010	K	E2
01010000	&	T	11010011	L	E3
01011010	!	E28	11010100	M	E4
01011011	$	E38	11010101	N	E5
01011100	*	E48	11010110	O	E6
01011101	)	E58	11010111	P	E7
01011110	;	E68	11011000	Q	E8
01011111	¬	E78	11011001	R	E9
01100000	-	E	11100010	S	02
01100001	/	0	11100011	T	03
01101011	,	038	11100100	U	04
01101100	%	048	11100101	V	05
01101101	_	058	11100110	W	06
01101110	>	068	11100111	X	07
01101111	?	078	11101000	Y	08
			11101001	Z	09
01111010	:	28			
01111011	#	38	11110000	0	0
01111100	@	48	11110001	1	1
01111101	'	58	11110010	2	2
01111110	=	68	11110011	3	3
01111111	"	78	11110100	4	4
			11110101	5	5
11000001	A	T1	11110110	6	6
11000010	B	T2	11110111	7	7
11000011	C	T3	11111000	8	8
11000100	D	T4	11111001	9	9
11000101	E	T5			

Note: T indicates a 12-zone punch.
E indicates an 11-zone punch.

Figure 5-13. 2560 Print Character Set

Note: Card movement is the same whether or not the Card Print special feature is installed. That is, the card ejected from the punch unit stops at the print station; the card then feeds into the stacker on the following cycle.

2560 Card Print Assembly

The card print assembly is available with two, four, or six print heads. The print heads are mounted below the card path so that the printing is done on the face of the card. Each print head can print a maximum of 64 characters on one horizontal line; spacing is 10 characters per inch. The centerline of the first printing position is 0.375 inch from the left edge of the card (the left edge of the first character is centered over column 2). *IBM Card Layout Form — Dual,* GX74-4049 and GX74-4555, provide for 10-to-the-inch spacing and can be used to design cards to be printed by the 2560.

Print heads can be set to print in 25 line positions, from above the 12-punch position to below the 9-punch position. The print heads, numbered 1 through 6, must remain in sequence from top to bottom, with print head 1 at the top. Therefore, with six print heads installed, print head 6 cannot be set above line 6 and print head 1 cannot be set below line 20.

Intermediate line positions are located on and between each row of punch positions. Print position 5 (between the 11-row and 0-row) should be avoided, if possible, because the feed wheel may cause some smudging of characters printed in that position. In punched fields, printing in even-numbered line positions should be avoided because punching may obliterate some characters. Figure 5-14 illustrates the style of printing and the available line positions.

2560 NOT-READY-TO-READY INTERRUPT—MODEL 15 ONLY

If interrupt level 6 is enabled, the 2560 sends an interrupt request to the system whenever the 2560 primary or secondary feed goes from a not-ready state to a ready state.

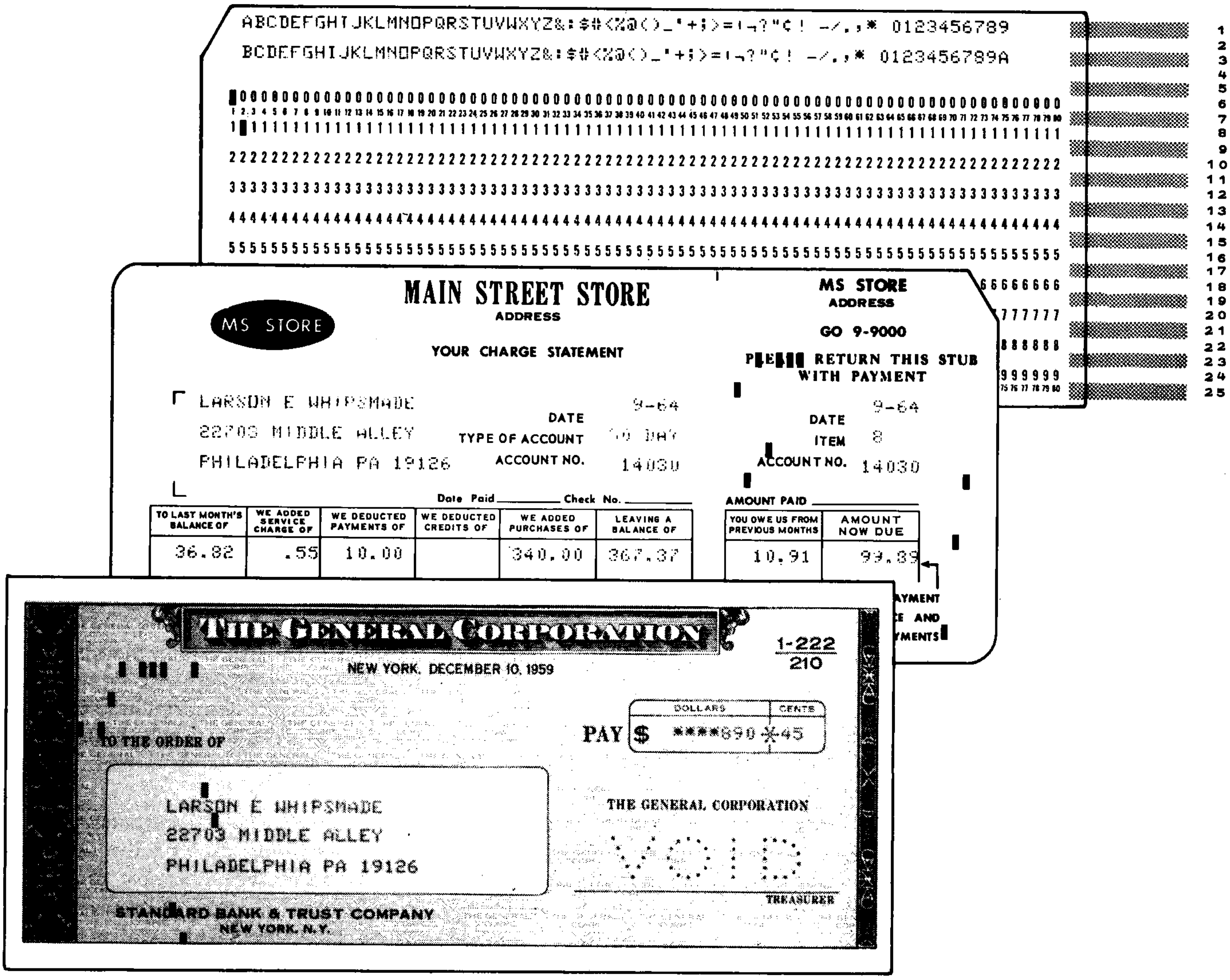


Figure 5-14. 2560 Card Document Printing

2560 OPERATOR CONTROLS

The operator controls (Figure 5-15) are located next to the secondary feed hopper. With these controls, the operator can monitor the operation of the 2560.

2560 PRIMARY and SECONDARY HOPPER CHECK Lights

The related light comes on when a card misfeeds from the primary hopper or the secondary hopper. To correct a card misfeed, remove the cards from the indicated hopper, repair or replace the damaged cards, replace the cards into the hopper, and press the START key. It is not necessary to clear the machine with the NPRO key.

2560 ATTENTION Light

This light comes on whenever:

- A stacker is full.
- The covers are open.
- The print interlock arm is not locked.
- The chipbox is full or out of position.
- The hand wheel is engaged or a hand crank is inserted.
- The CE EMERGENCY STOP switch is in the stop position.

The operator must correct any of these conditions before the primary or secondary feed can return to a ready condition.

The ATTENTION light turns off both ready lights and turns on the I/O CHECK light on the processing unit console.

2560 POWER ON Light

This light turns on when power is applied to the control circuits of the 2560.

2560 FEED CHECK Light

This light and the associated feed-check sense bit comes on (1) when a card is mispositioned in the card path, (2) when the covers are opened while the 2560 is operating (the ATTENTION light also comes on), or (3) when certain equipment malfunctions occur.

To turn off the FEED CHECK light when it has turned on because of a mispositioned card, (1) empty both hoppers, (2) open the covers, (3) clear the card paths, (4) close the covers, and (5) press the NPRO key.

When the FEED CHECK light comes on, the PRIMARY and SECONDARY READY lights turn off and the I/O CHECK light on the CE panel turns on.

2560 MACHINE CHECK Light

The MACHINE CHECK light indicates that the 2560 (1) went through an extra clutch cycle, (2) failed to write on the read-emitter drum, (3) failed to erase the read-emitter drum, or (4) detected an extra or missing feed, punch or print CB pulse. The machine check sense bit also turns on whenever the MACHINE CHECK light comes on.

To turn off the MACHINE CHECK light, (1) empty both hoppers, (2) open the covers, (3) clear the card paths, (4) close the covers, and (5) press the NPRO key. If the trouble persists, notify the customer engineer.

The cause of the MACHINE CHECK light coming on also turns off the PRIMARY and SECONDARY READY lights and turns on the I/O CHECK light on the CE panel.

2560 READ CHECK and PUNCH CHECK Lights

Detection of an error or an invalid code during reading turns on the READ CHECK light. An invalid code transfers to main storage as a hex 00. Detection of an error during punching turns on the PUNCH CHECK light. These lights are turned off (1) by the program-sensing of the read/punch check bit followed an SIO instruction, (2) by operation of the CHECK RESET key on the processing unit console, or (3) by operation of the NPRO key on the 2560. The I/O CHECK light on the CE panel always comes on when either the READ CHECK or PUNCH CHECK light comes on.

2560 PRIMARY and SECONDARY READY Lights

The related light indicates that the primary feed or the secondary feed can accept instructions from the processing unit. The conditions required of each feed for the light to be on are:

- Power on
- A card in the related preread station, except during last-card sequences
- Cards in the appropriate hopper, except during last-card sequences
- None of the following error lights on:
  - Primary hopper check
  - Secondary hopper check
  - Attention
  - Feed check
  - Machine check
- Machine not stopped with the STOP key

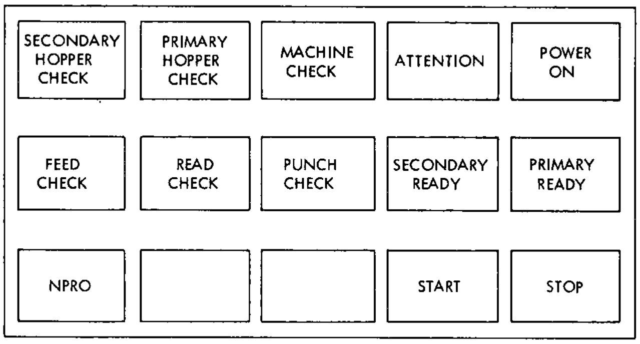


Figure 5-15. 2560 Operator Control Panel

2560 START Key

Pressing START turns on the primary and secondary ready lights if the required conditions are met. For a run-in, pressing the start key (1) feeds a card from the secondary hopper to the secondary preread station and (2) feeds two cards from the primary hopper: one to the primary preread station and one to the primary input station. The primary and secondary feeds are independent; that is, either feed can operate without the other.

The start key is also used to restore the primary or secondary ready status after a hopper check condition.

2560 STOP Key

Pressing STOP stops the 2560 and removes it from a ready status. Any reading, punching, or printing operation in process is completed before the machine stops.

2560 NPRO Key

Pressing the NPRO key starts a series of six feed cycles that run out all cards from both the primary and the secondary card paths without processing the cards. Cards are automatically stacked normally: that is, primary cards into stacker 1 and secondary cards into the highest numbered stacker installed.

The NPRO key is not effective unless the primary and secondary hoppers are empty and the ATTENTION light is off.

If the FEED CHECK light is on and the NPRO key is inoperative, remove any jammed cards from the stacker transport area and the cards from the punch unit and prepunch stations.

The NPRO key should be pressed just before any program is loaded and again whenever any job is finished. This clears the machine of any cards from a previous job or the job you currently processed.

PROCESSING UNIT LIGHTS ASSOCIATED WITH 2560

I/O CHECK Light

This light turns on immediately when one of the following conditions occurs in the 2560:

Primary hopper check
Secondary hopper check
Feed check
Read check
Punch check
Print check
Machine check
Adapter check
No-op bit is set

I/O ATTENTION Light

This light is turned on by an SIO instruction being directed to the 2560 when the ready light for the specified feed is not on.

2560 Billable Time Metering

Each 2560 is equipped with a meter that records billable time for the 2560 while the system is in operation and the 2560 is online and operational. The 2560 use meter runs while all of the following conditions apply:

- There is at least one card in the 2560 transport.
- The 2560 has accepted a command since a card runout condition last occurred.
- The system's processing unit is initiating, executing, or completing an instruction or command (including an I/O or assignable unit instruction or command).

2560 OPERATING PROCEDURES AND TIMINGS

The operating procedures and timing considerations for the 2560 are discussed in detail in *IBM System/360 and System/370 Component Description and Operating Procedures: IBM 2560 Multi-Function Card Machine,* GA21-5893, which is available from your IBM representative or the branch office serving your locality.

2560 Card Read Operations

The read/feed functions of start I/O instructions move a card from the specified hopper to the corresponding input or preread station. At the same time, each card in the card path is advanced to the next position. If read is specified, the data contained in the card passing the read station is transferred to storage at a field specified by a load I/O instruction. The read data is checked to ensure that it is read correctly. An error in reading causes a read check and data of hex 00 is transferred to the processing unit for the column in error.

Two load I/O instructions must be executed before a start I/O instruction that specifies card reading. These load I/O instructions must load the boundary (lowest) address of the read data field into the 2560 read data address register and the length count of the read data into a length count register. To meet specifications, the address must be on a 128-byte boundary (000, 128, 256, etc). An SIO read command issued with a read data length count of 0 sets the no-op status bit and the SIO is not executed. An SIO issued with a data length count greater than maximum (hex 50) is executed with the maximum length count.

The card feeding and reading rate is determined by the operations being performed. The rated reading speeds (500 cards per minute for Model A1 and 310 cards per minute for Model A2) are for read operations only.

2560 IPL Read

Pressing the PROGRAM LOAD key on the processing unit with the PROGRAM LOAD switch set to ALTERNATE causes the following reader actions to occur:

- The MFCM read data address register is set to 0000.
- A read operation is performed from the primary hopper of the MFCM without a start I/O instruction being executed.

The IPL read operation is performed similar to a start I/O card read operation.

2560 Punch Operations

Start I/O instructions that specify punching initiate moving a card from one of the prepunch stations (primary or secondary), through the punch station. Data from storage is recorded in the card in the form of punched holes. The punching is checked to ensure that the correct data is punched. An error causes a punch check.

Two load I/O instructions must be executed before a start I/O instruction that specifies a punch operation. These load I/O instructions place the boundary (lowest) address of the punch data field into the MFCM punch data address register and the length count of the punch data into a length count register. Column 1 of the card is punched with the data contained in storage at the address specified by the MFCM punch data address register. Column 2 of the card is punched with the data contained in storage at the next higher address. The punch data fields must be on a 128-byte boundary similar to the read operation.

If a punch start I/O instruction is given with no card in the prepunch station or with a length count of 0 the no-op status bit will be set and the SIO is not executed. A data length count in excess of the maximum (hex 50) causes the instruction to execute using the maximum data length.

Card punching is performed at a single rate for each model of 2560, Model A1 at 160 columns per second and Model A2 at 120 columns per second.

2560 Print Operations

The start I/O print instruction initiates card motion from the preprint station through the print station where up to six lines of up to 64 characters, each may be printed on the card. A print overrun or a print translator parity error during the print operation sets the print check status bit and turns on the I/O CHECK light on the CPU console.

Before a start I/O print instruction is executed three operations must be completed: (1) the boundary (lowest) address of the print data area must be loaded into the print data address register using a load I/O instruction; (2) the print data length count and print heads (lines) to be used must be set into registers using a single load I/O instruction; and (3) the print data area must contain the data to be printed.

The MFCM with a Print feature (available on Model A1 only) installed is configured to either two, four or six print lines, and any or all of these print lines may be used at one time on a given card. All of the print lines use the same length count. The length count must equal the largest number of characters to be printed on any given line, and all shorter lines must be filled with some character (usually hex 40) to equal the length count specified. Data areas for unselected print lines are ignored and are not required to be filled.

The print data area must start on a 256-byte boundary (000, 256, 512, etc). An area of 64 bytes must be allotted for each print line up to the highest numbered line being printed:

Line 1 — Print data address to byte 64
Line 2 — Bytes 65 through 128
Line 3 — Bytes 129 through 192
Line 4 — Bytes 193 through 256
Line 5 — Bytes 257 through 320
Line 6 — Bytes 321 through 384

If, for example, printing is being done on lines 1 and 4 only, the storage area allotted must include area for lines 1, 2, 3, and 4. The data to be printed must be placed within the line area allocation starting at the lowest address of that area. The data for line 1 in the example would start at the address contained in the 2560 print data address register and the data for line 4 would start 193 bytes higher in storage.

If (1) no head is selected, (2) a print data length count of zero is specified, or (3) a start I/O print command is issued with no card in the preprint station, the 2560 sets the no-op status bit on and the SIO is not executed. An SIO print command does not include a feed cycle; two consecutive print commands without an intervening feed or read command sets the no-op status bit on. This occurs because there is no card in the preprint station for the second print command.

The MFCM prints any of the 64 characters in the card code. The rated throughput in printing operations is 140 characters per second.

2560 Combined/Overlapped Operations

When a combined command is issued (for example, print—punch—read), the printing is started first, then the punching is started, and both are performed together. At the completion of printing and punching, a feed cycle is initiated (in our example) and the reading is performed during this feed cycle.

Overlapping occurs when a punch-feed or punch-read command is issued while a print command is being executed. Thereafter, the remainder of the operation resembles a combined command operation.

Separate print and punch commands are not overlapped if the punch command is executed first. When an SIO punch command is executed with a card at the print station, the print card is ejected through the print station in order to prevent a card jam. The print command, issued after the punch command, is not executed until the card being ejected leaves the print station and the next card is registered in the print station.

Maximum throughput is realized when the commands are issued in an order that allows the 2560 to operate in an overlapped mode.

2560 Input/Output Timing

There are two basic timing considerations of importance to a user of an IBM 2560 Multi-Function Card Machine:

- Card throughput in cards per minute (cpm)
- Time available for other system operations

2560 Card Throughput

Optimum usage of the 2560 is obtained by maintaining the maximum rated throughput. In order to accomplish this, all input/output instructions should be programmed to take advantage of continuous reading or overlapped operations. The following sections explain and demonstrate each of these.

2560 Basic Operating Times

There are three basic operating times: reading, punching, and printing.

2560 Reading Times

The 2560-A1 requires 120 milliseconds and the 2560-A2 requires 200 milliseconds to read one card, regardless of the number of columns read. The maximum throughput of 500 cards per minute for the Model A1 and 310 cards per minute for the Model A2 is based on continuous operation. This means that the sequence and timing of the program instructions must not allow the feed clutch to disengage.

Figure 5-16 shows the timing breakdown of a feed cycle. Note that the clutch decision point is 16 ms for Model A1 (25 ms for Model A2) before the end of the cycle. If the next read or feed instruction occurs after the clutch decision point, the clutch disengages; 16 ms for Model A1 (25 ms for Model A2) must be added to the total feed cycle time to allow the clutch to be re-energized and re-engaged. Therefore, to maintain continuous operation of the feed cycles, the next read or feed instruction must occur before the clutch decision point.

2560 Punching Times

The time required to punch a card is determined by the number of columns specified by the instruction. Spacing over a column for which no data is sent from the processing unit takes the same amount of time as punching; thus, blank columns do not affect the punching rate. The time required for the 2560 Model A1 to punch one card is expressed by the formula:

T(ms) = 6.11 (LColP) + 170

where LColP = last column punched.

The formula for the 2560 Model A2 is:

T(ms) = 8.33 (LColP) + 263

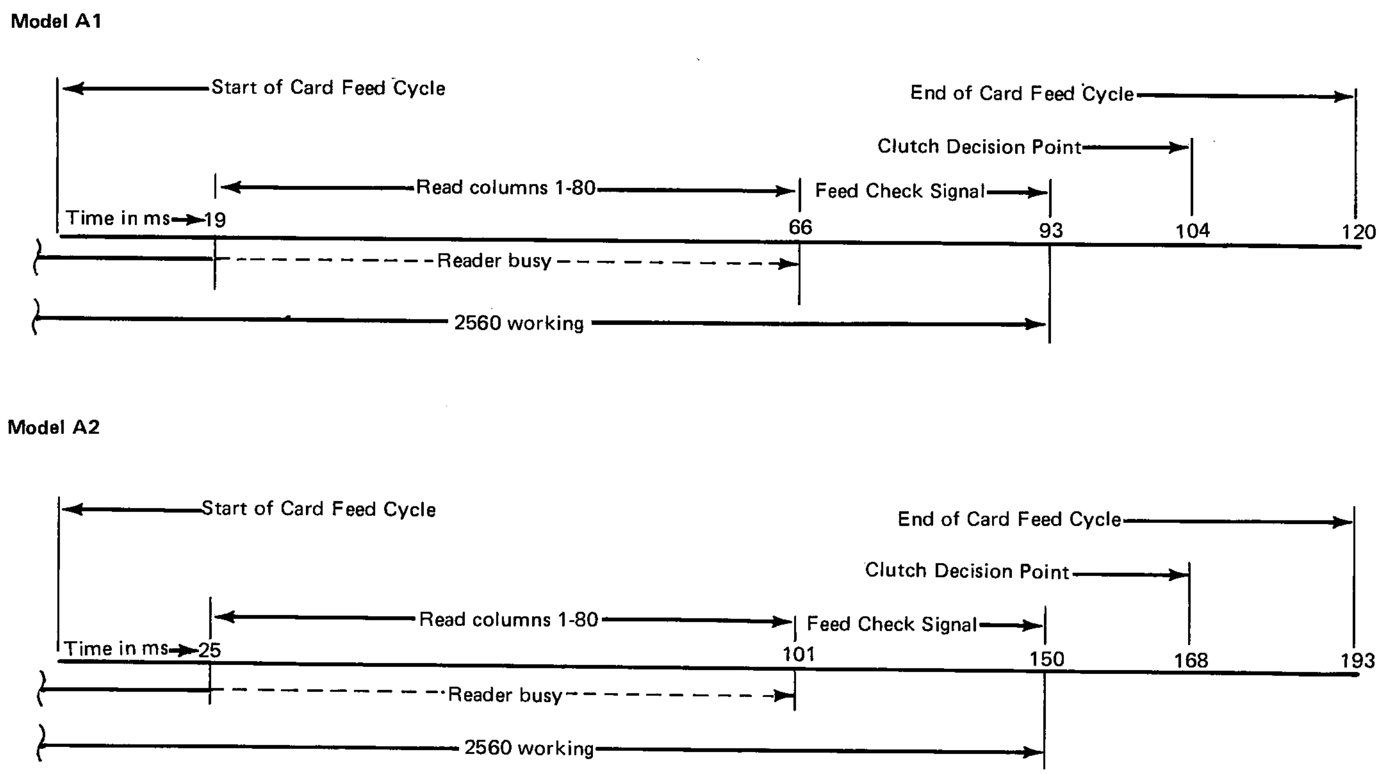


Figure 5-16. 2560 Read Cycle Timing

The timing relationships of the punch cycles are shown in Figure 5-17. The 34 ms of the Model A1 formula and the 54 ms of the Model A2 formula represent card registration times. Clutch pickup time is not necessary after a punch instruction because punching requires a feed cycle. The card feed cycle or eject cycle follows each punch operation.

Punch-and-Feed or Punch-and-Read Instruction: The card-feed cycle occurs immediately after punching and clutch pickup, as shown in Figure 5-18. All cards in the feed designated by the instruction and those cards beyond the prepunch station move to the next station.

The significant timing characteristic of a punch and read or punch and feed instruction is that the card cycle occurs before the next instruction is received, when punching is completed.

2560 Printing Times

The time required to print one card is determined by the number of characters specified by the instruction. Character positions not printed have no effect on the printing rate because spacing over a position takes the same amount of time as printing the character. The time required to print a card is expressed by the formula:

$$T(ms) = 7.23(LChP) + 136$$

where LChP = last character printed

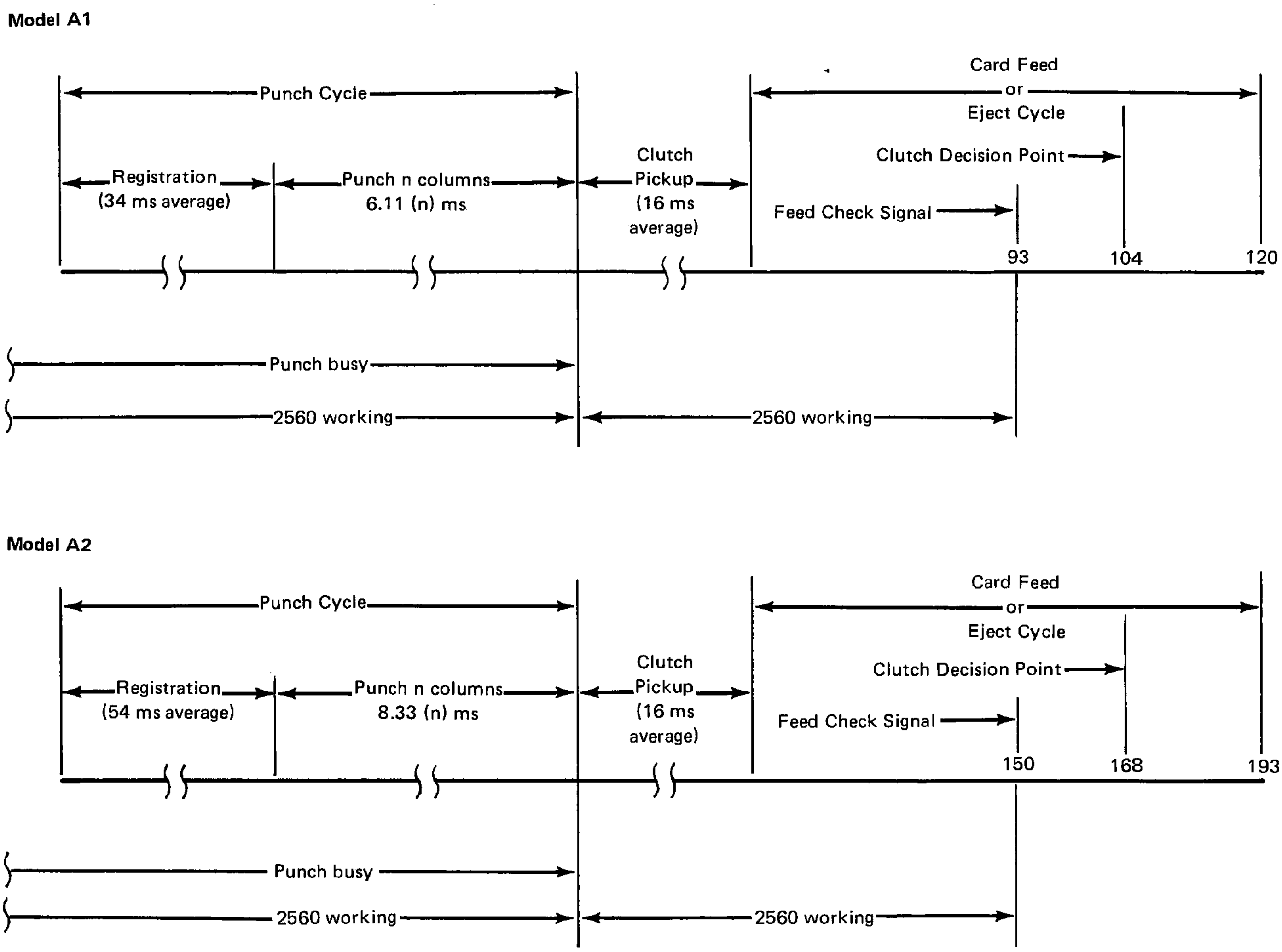


Figure 5-17. 2560 Punch or Punch-and-Feed Cycle Timings

Figure 5-17 shows the timing relationships of the print cycle. Note that punching can begin after the fourth character has been printed, about 28 ms after printing begins. This is important to the print-punch sequence.

Card throughput for punching or printing is shown in Figure 5-18.

2560 Combined Operation Timings

Seldom does an application consist of just one operation. In fact as noted previously, successive SIO print instructions cause the no-op status bit to be set on and must therefore, be used in combination with at least one of the SIO instructions that generate a feed cycle.

The following paragraphs describe operations that can be overlapped and save considerable time in processing a card file.

2560 Overlapped Punch-Read Operations (Same Feed): The formula for computing punching time assumes that the 2560 goes through a 120 ms cycle for Model A1 (200 ms for Model A2) after each punch instruction (see Figure 5-17). When a punch-read instruction is issued, the read supplies the card cycle and, in effect, overlaps the feed cycle portion of the punching time expressed by the formula. Thus, both operations can be performed in the time derived from the punch formula.

2560 Overlapped Print-Punch Operation Timing: Printing and punching are overlapped in the 2560 if the print instruction is given first and followed immediately by a punch and read instruction or a punch and feed instruction. The printing operation starts first. Punching begins after about the fourth character position has been printed or skipped.

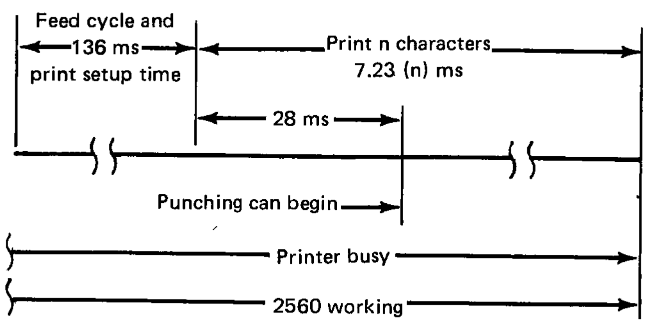


Figure 5-18. 2560 Write Cycle Timing

Punching			Printing[1]	
Last Column Punched	Approximate Cards per Minute (cpm)		Last Character Printed	Approximate Cards per Minute (cpm)
	A1	A2		
1	340	221	1	395
5	300	196	5	335
10	260	173	10	275
15	230	154	15	240
20	205	140	20	210
25	185	127	25	185
30	170	116	30	165
35	155	108	35	150
40	145	100	40	140
45	135	94	45	128
50	126	88	50	120
55	119	83	55	110
60	112	79	60	105
70	100	70	64	100
80	91	65		

Formula for card punching (approximate cards/minute):

$$\text{CPM, Mod A1} = \frac{60{,}000}{6.11\ (\text{LColP}) + 170}$$

$$\text{CPM, Mod A2} = \frac{60{,}000}{8.33\ (\text{LColP}) + 263}$$

Formula for card printing (approximate cards/minute):

$$\text{CPM} = \frac{60{,}000}{7.23\ (\text{LChP}) + 136}$$

[1]Model A1 only

Figure 5-19. 2560 Approximate Card Throughput for Punching or Printing

The time required for punching and printing can be determined by using the longer time of the following two formulas:

$$T(\text{ms}) = 6.11\ (\text{LColP}) + 170\ (\text{punching})$$

or

$$T(\text{ms}) = 7.23\ (\text{LChP}) + 136\ (\text{printing})$$

2560 SAMPLE PROGRAM

The sample program (Figure 5-20) demonstrates the use of overlapped instructions, such as block 6 (punch-read) and blocks 10 and 14 (print) followed by a punch and read (block 6).

Assume that 12,000 transaction cards are key-punched from invoices or sales slips, as follows:

Item Code	7 cols
Quantity	4 cols
Price	6 cols
Date	5 cols
Customer Code	3 cols

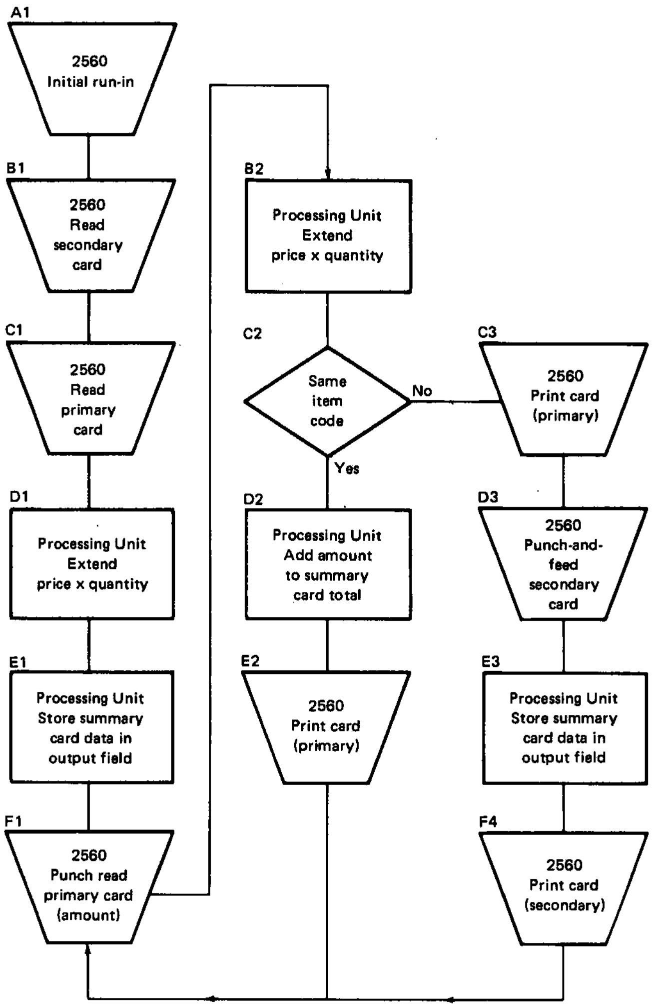


Figure 5-20. 2560 Program Block Diagram (Sample Program)

The amount field (the price times the quantity) is punched in columns 1 through 7 by the 2560. All columns (1 through 33) are printed. There is an average of three transaction cards for each item code. A summary card is punched by item code and contains all the data, except customer code. All columns 1 through 30 are printed on the summary card.

Figure 5-21 shows how to compute the total processing time. Numbers to the left of the operation identify the block in the block diagram in Figure 5-20. Time in the column on the right is in minutes. The total processing time of 99 minutes saves 67 minutes (40 percent) when compared with the time required if the print and punch-read sequences are not used.

Block No.	Operation and Formula	Minutes
10	Print 8,000 transaction cards (7.23 x 33) + 136 x 8,000 =	50.0
6	Punch read 8,000 transaction cards (6.11 x 8) + 170 x 8,000 =	
11	Print 4,000 transaction cards (7.23 x 33) + 136 x 4,000 =	25.0
12	Punch and feed 4,000 summary cards (6.11 x 30) + 170 x 4,000 =	
14	Print 4,000 summary cards (7.23 x 30) + 136 x 4,000 = Punch read 4,000 transaction cards (6.11 x 8) + 170 x 4,000 =	23.6
	Total processing time	98.6

Figure 5-21. 2560 Program Processing Time Calculation (Sample Program)

Figure 5-22 shows each 2560 operation by successive moves of cards through the machine. Cards are identified by a P for primary feed (transaction) cards and an S for secondary feed (summary) cards. The numbers establish the sequence of the cards. The corresponding blocks of the block diagram (Figure 5-20) define the instructions.

Block	Operation	Positions of Cards at End of Operation						
			Read Station					
		Primary Input Station	Preread Station Primary Secondary	Prepunch Station Primary Secondary	Punch Station	Print Station	Printed	Stacked
1	Initial run-in	P2	P1 S1					
2	*Read secondary card* First blank card (S1) is read to move it into the prepunch station.	P2	P1 S2	S1				
3	*Read primary card* First transaction card (P1) is read.	P3	P2 S2	P1 S1				
6	*Punch primary card* Amount computed and stored in blocks 4 and 5 is punched into first transaction card. *Read primary card* Second transaction card (P2) is read; first transaction card (P1) advances to and is registered at the print station.	P4	P3 S2	P2 S1		P1		
11	*Write card* First transaction card is printed, positions 1 through 33.	P4	P3 S2	P2 S1			P1	
6	*Punch primary card* Amount computed and stored in blocks 8 and 11 punched into second transaction card. *Read primary card* Third transaction card (P3) is read. P1 goes into stacker 1; P2 advances to and is registered at print station.	P5	P4 S2	P3 S1		P2		P1
11	*Write card* Second transaction card is printed, positions 1 through 33.	P5	P4 S2	P3 S1			P2	P1
12	*Punch and feed secondary card* Third transaction card is the first card of new group. Summary card (S1) for first group first and second transaction cards is punched. Punch and feed instruction moves blank card (S2) into prepunch station and first summary card to print station. Feed cycle puts P2 into stacker 1.	P5	P4 S3	P3 S2		S1		P2 P1

Figure 5-22 (Part 1 of 2). 2560 Card Movement (Sample Program)

		Positions of Cards at End of Operation							
			Read Station						
Block	Operation	Primary Input Station	Preread Station Primary Secondary	Prepunch Station Primary Secondary	Punch Station	Print Station	Printed	Stacked
14	*Write card* First summary card is printed, positions 1 through 30.	P5	P4 S3	P3 S2			S1	P2 P1
6	*Punch primary card* Amount computed and stored in blocks 8 and 14 is punched into third transaction card. *Read primary card* Fourth transaction card (P4) is read. Summary card (S1) is ejected into stacker. P3 advances to and is registered at the print station.	P6	P5 S3	P4 S2		P3		P2 P1 S1

Figure 5-22 (Part 2 of 2). 2560 Card Movement (Sample Program)

2560 START I/O (SIO)

Op Code (hex)	Q-Byte (binary)	R-Byte (binary)
Byte 1	Byte 2	Byte 3
F3	1111 x xxx	xxxx xxxx

DA M N Control Code

Bits[1]

0123 4567	Function Specified
xxxx x000	Stack primary card in stacker 1; stack secondary card in stacker 5 (if 2560 is Model A1) or stacker 4 (if 2560 is Model A2) (these are default stackers).
xxxx x001	Select stacker 1
xxxx x010	Select stacker 2
xxxx x011	Select stacker 3
xxxx x100	Select stacker 4
xxxx x101	Select stacker 5 if 2560 is Model A1; default to stacker 4 if 2560 is Model A2
00xx xxxx	Disable interrupts
01xx xxxx	Enable interrupts
10xx xxxx	Reset interrupts
11xx xxxx	Reset/enable interrupts

N-Code	Operation
000	Feed
001	Read
010	Punch and feed
011	Punch and read
100	Print, no feed (M-code unused; cards do not feed)
101	Interrupt control (M-code unused; cards do not feed)
110	Print, punch, and feed
111	Print, punch, and read

0 = Perform operation on card in primary feed
1 = Perform operation on card in secondary feed

Hex F specifies the 2560 as the device to be controlled.

F3 specifies a start I/O operation. F as the first hex character in the op code specifies a command-type instruction (that is, an instruction with no operand addressing).

[1] Bits 2, 3, and 4 are not used; they should be 0. Bits 1 and 2 (for interrupt control) are ignored for all SIO instructions that cause card movement. Bits 5, 6, and 7 (stacker select control) are ignored for SIO interrupt control instructions.

Operation

The 2560 performs the functions specified by the N-code and control code on the card in the feed path specified by the M-code. Stacker selection applies to the card at the prepunch station when this instruction is executed.

Program Notes

- Stacker selection control bits (5, 6, and 7 of the control code) of 000, 110, and 111 send cards to the default stackers. A control code of 101 on a 2560 Model A2 will send the card to stacker 4. All stacker selection is performed for the card in the prepunch station. If there is no card in the prepunch station, the stacker select information is ignored.

- The attachment aborts the instruction and sets the no-op status bit, and requests an op-end interrupt under the following conditions:
  - No card is available in the transport for the requested function. *Exception:* a read with no card available forces a feed cycle (instead of a no-op function) because this condition occurs only on last card routines.
  - The program issues an SIO when the attachment has indicated a machine check or feed check, and the check has not been resolved, or when the sense bytes contain an unsensed no-op or data check bit.
  - The program issues an SIO and the length count loaded (via an LIO) for that data operation equals zero.[1]
  - The program issues a print SIO with zero heads selected (via an LIO).

- If the 2560 is busy or requires operation intervention (processing unit I/O ATTENTION light is on) when the program issues the SIO instruction, the program loops on the SIO until the condition is no longer present.

[1] This condition does not cause a no-op for a print SIO if the 2560 is not equipped with the print feature. Instead, the 2560 accepts the instruction and performs a print operation with the nonexistent feature.

- If the program issues the SIO when the not-ready condition exists, the I/O attention light on the processing unit turns on.

- For read commands and feed commands, the attachment ignores stacker control information unless a card is in the prepunch station when the instruction is accepted.

- An interrupt control SIO is unconditionally accepted by the 2560.

2560 Op End Interrupts

The 2560 operates on interrupt level 5. It presents an op end interrupt request to the processing unit at the end of execution of a start I/O command (whether execution stopped normally or because of an error or no-op condition on the 2560). Interrupts can be enabled, disabled, or reset at any time. The program can use the TIO instruction to test for interrupt enabled and interrupt pending conditions. If an interrupt is pending, the program can determine the reason for the interrupt request by means of the sense instruction.

2560 TEST I/O AND BRANCH (TIO)

Op Code (hex)	Q-Byte[1] (binary)	Operand Address	
Byte 1	Byte 2	Byte 3	Byte 4
C1	1111 x xxx	Operand 1 address	
D1	1111 x xxx	Op 1 disp from XR1	
E1	1111 x xxx	Op 1 disp from XR2	

DA M N

N-Code	Condition Tested
000	Feed not ready or error
001	Ready busy[2]
010	Punch busy[2]
011	Any busy (including feed)[2]
100	Print busy[2]
101	Interrupts enabled
110	Either punch busy or print busy[2]
111	Interrupts pending

M: 0 = Primary feed when N = 000; must be used when N is not 000
1 = Secondary feed when N = 000; invalid when N is not 000

DA: Hex F specifies the 2560 as the addressed device.

Hex C1, D1, or E1 specifies a test I/O and branch instruction. The first hex character in the op code signifies the type of operand addressing to be used by the instruction.

---

[1] A Q-byte of F9 through FF is invalid and results in a program check if interrupt level 7 is enabled or a processor check if interrupt level 7 is not enabled.

[2] If a feed check, machine check, or hopper check occurs when any busy condition exists, the busy indication is turned off.

Operation

The processing unit tests the 2560 feed specified by the M-code for the condition specified by the N-code. If the condition exists, the program branches to the address in the operand portion of the instruction. If the condition does not exist, the program immediately accesses the next sequential instruction.

Program Notes

If a branch occurs, the address recall register will contain the address of the next sequential instruction and the instruction address register will contain the branch-to address at the end of the TIO operation.

2560 ADVANCE PROGRAM LEVEL (APL)

Op Code (hex)	Q-Byte[1] (binary)	R-Byte (binary)
Byte 1	Byte 2	Byte 3
F1	1111 x xxx	0000 0000

DA M N — R-byte is not used in an APL instruction

N-Code	Condition Tested
000	Feed not ready or error
001	Read busy[2]
010	Punch busy[2]
011	Any busy (including feed)[2]
100	Print busy[2]
101	Interrupts enabled
110	Either punch busy or print busy[2]
111	Interrupts pending

M:
0 = Primary feed when N = 000; must be used when N is not 000
1 = Secondary feed when N = 000; invalid when N is not 000

DA: Hex F specifies the 2560 as the addressed device.

Hex F1 specifies the advance program level instruction. F as the first hex character in the op code specifies a command-type instruction (that is, an instruction with no operand addressing)

---

[1] A Q-byte of F9 through FF is invalid and results in a program check if interrupt level 7 is enabled or a processor check if interrupt level 7 is not enabled.
[2] If a feed check, machine check, or hopper check occurs when any busy condition exists, the busy indication is turned off.

Operation

This instruction tests for the conditions specified in the Q-byte.

- Condition present:
  - Systems with Dual Program Feature installed and enabled, activate the inactive program level.
  - Systems without Dual Program Feature installed or with Dual Program Feature installed but not enabled, loop on the advance program level instruction until the condition no longer exists.

- Condition not present: Systems with or without Dual Program Feature access the next sequential instruction in the active program level.

Program Note

For additional information concerning the advance program level instruction, see Chapter 2.

2560 LOAD I/O (LIO)

Op Code (hex)	Q-Byte (binary)	Operand Address	
Byte 1	Byte 2	Byte 3	Byte 4
31	1111 x xxx	Operand 1 address	
71	1111 x xxx	Op 1 disp from XR1	
B1	xxxx x xxx	Op 1 disp from XR2	

DA M N

N-Code **Register and Data to be Loaded**

000 Op address byte: Read data length count (X'50' max)
Op address -1 byte: Not used

001 CE diagnostics

010 Op address byte: Not used
Op address -1 byte:. Punch data length count (X'50' max)

011 Op address byte: Print head select control data:

Bit[1]

0123 4567	**Meaning**
xx00 0000	Invalid (instruction is aborted and no-op bit turns on)
xxxx xxx1	Select print head 1
xxxx xx1x	Select print head 2
xxxx x1xx	Select print head 3
xxxx 1xxx	Select print head 4
xxx1 xxxx	Select print head 5
xx1x xxxx	Select print head 6

Op address -1 byte: Print data length count (X'40' max)

100 The data located at the operand and operand -1 addresses is loaded into the MFCM print data address register.

101 The data located at the operand and operand -1 addresses is loaded into the MFCM read data address register.

110 The data located at the operand and operand -1 addresses is loaded into the 2560 punch data address register.

111 Invalid

An invalid N-code causes one of the following conditions to occur:
Processor check if the check occurs in interrupt level 7 while interrupt is not enabled.
Program check if the check occurs while processor check interrupt level 7 is enabled.

M: 0 = Normal operating mode
1 = CE diagnostic mode

DA: Hex F specifies the 2560 as the addressed device.

Hex 31, 71, or B1 specifies a load I/O operation. The first hex digit designates the type of operand addressing to be used for the instruction.

[1]Bits 0 and 1 are not used; they should be 00. Bits 2-7 can be set on in any desired combination to select any combination of print heads.

Operation

The processing unit moves the contents of the 2-byte field specified by the operand address to the register specified by the N-code.

Program Notes

- If the program issues an SIO for which a length count of 0 has been established, the instruction is aborted and the no-op sense bit turned on. A print SIO with a length count of 0 or no heads selected issued to a 2560 without the print feature installed is accepted and performed with the nonexistent feature instead of being aborted.

- Length counts exceeding the maximum length counts are accepted as the maximum values (hex 40 for print operations; hex 50 for read and punch operations).

- Because the read data length and punch data length LIO instructions each operate with a single byte of data (byte 1, the byte located at the operand address, contains the read data length count, while byte 2, the byte located at the operand address minus 1, contains the punch data length count), the same CPU main storage address can be used for read and punch LIO instructions.

- The read and punch data areas must each start on a boundary that is a multiple of 128 (000, 128, 256, etc).

- The print data area must start on a boundary that is a multiple of 256 (000, 256, 512, etc). A 64-position subarea must be allotted for each print head selected up to the highest numbered head selected. If, for example, print head 3 is the only head being used, then the data area should start on the boundary specified previously and be 128 positions (64 each for heads 1 and 2) plus the length count specified for head 3 data. The data areas are in consecutive 64-position increments with head 1 being in the area with the lowest address and head 6 (if available) at the highest address. The highest selected head requires a data area length equal to the specified length count, however, the full 64 positions may be allotted if so desired.

- There is only one length count specified for all the print lines being used. When required each line selected must have its subarea filled (usually with hex 40) so the total number of characters in each print line is equal to the specified length count. Data areas beyond the specified length count or for nonselected heads need not be filled as they are ignored.

2560 SENSE I/O (SNS)

Op Code (hex)	Q-Byte (binary)	Operand Address	
Byte 1	Byte 2	Byte 3	Byte 4
30	xxxx x xxx	Operand 1 address	
70	xxxx x xxx	Op 1 disp from XR1	
B0	1111 0 xxx	Op 1 disp from XR2	

DA M N

N-Code **Data Being Sensed (M=0 only)**

EB1 = Operand Address, EB2 = Operand Address -1

N-Code	Data Being Sensed
000	EB1: Register address at time of adapter check EB2: Adapter checks
001	With SNS N-code 011 EB1 bit 1 on (machine check) EB1: Machine checks EB2: Data checks
001	With SNS N-code 011 EB1 bit 2 on (feed check) EB1: Machine checks EB2: Data checks
001	With SNS N-code 011 EB1 bit 3 on (data check) EB1: Data column in error EB2: Data checks
001	With SNS N-code 011 EB1 bits 1, 2, and 3 off EB1: Number of columns punched on last punch command EB2: Data checks
010	With SNS N-code 011 EB1 bit 2 on (feed check) EB1: Feed checks EB2: Feed checks
010	With SNS N-code 011 EB1 bit 3 on (data check) EB1: Rows 4-9 in error or raw data on a read validity error EB2: Rows 12-3 in error or raw data on a read validity error
010	With SNS N-code 011 EB1 bits 2 and 3 off EB1: Number of columns printed on last print command EB2: Number of columns read on last read command
011	EB1: 2560 general indicators EB2: Restart byte (card positioning)
100	EB1 and EB2: Print data address register
101	EB1 and EB2: Read data address register
110	EB1 and EB2: Punch data address register
111	EB1 and EB2: Invalid N-code

M:
- 0 Normal mode
- 1 Diagnostic mode (for CE use)

Hex F specifies the 2560 as the addressed device.

30, 70, or B0 specifies a sense I/O operation. The first hex character in the op code specifies the type of operand addressing for the instruction.

Operation

The CPU transfers 2 bytes of data from the unit specified by the N-code to the main storage field specified by the operand address. The first byte transferred enters the effective address (the operand address), the second byte enters the effective address minus 1. Status bits are described in Figure 5-23.

N-Code	EB2	EB1
SNS X'000'	— Reserved	— Reserved
	1 — CSAR check	1 —
	2 — Control store check	2 —
	3 — ALU check	3 — MS/LS/EXT register
	4 — X-reg check	4 — address frozen on
	5 — ALU bus check	5 — adapter check
	6 — X-reg ms check	6 —
	7 — Y-reg check	7 —
SNS X'001'	0 — Read overrun	0 — Overlap bit[1]
	1 — Punch overrun	1 — Column emitter read/write check
	2 — Print overrun	2 — Columns emitter erase check
A Machine	3 — Read compare	3 — Extra feed cycle check
Check	4 — Punch compare	4 — Feed CB sequence check
	5 — Invalid read character	5 — Punch push extra cycle check
	6 — Fiber optic check	6 — Punch/increment CB sequence check
	7 — Print translate check	7 — Print CB sequence check
SNS X'001'	0 — Read overrun	0 — Overlap bit[1]
	1 — Punch overrun	1 —
	2 — Print overrun	2 —
0 Data Check	3 — Read compare	3 —
	4 — Punch compare	4 — Specifies data column in error
	5 — Invalid read character	5 —
	6 — Fiber optic check	6 —
	7 — Print translate check	7 —
SNS X'010'	0 — Input station check	0 — Read station late check
	1 — Preread primary check	1 — Punch station check
	2 — Prepunch primary check	2 — Print station check
B Feed Checks	3 — Punch pusher primary check	3 — Cell 8 to 9 check
	4 — Preread secondary check	4 — Corner station check
	5 — Prepunch secondary check	5 — Jam bar check
	6 — Punch pusher secondary check	6 — Cover interlock check
	7 — Read station early check	7 — Reserved
SNS X'010'	0 — Compare high	0 — Compare high
	1 — Compare low	1 — Compare low
	2 — 12	2 — 4
0 Data Checks	3 — 11	3 — 5
	4 — 0 Rows in error or raw data on a	4 — 6 Rows in error or raw data on a
	5 — 1 read validity check	5 — 7 read velocity check
	6 — 2	6 — 8
	7 — 3	7 — 9
SNS X'011'	0 — Card at input station	0 — Primary last card indicator
	1 — Card at secondary preread	1 — Machine check A
	2 — Card at primary preread	2 — Feed check B
	3 — Card at secondary prepunch	3 — Any data check C
	4 — Card at primary prepunch	4 — Secondary last card indicator
	5 — Card at print station	5 — No-op
	6 — Reserved	6 — Primary hopper check
	7 — Reserved	7 — Secondary hopper check

[1] If set to 1, an overlapped operation (print/punch) occurred.

SENSE N-code 011 EB2 will have been updated to reflect the position of cards after the operation if the error is a soft type (data check) and will reflect the position of cards prior to the operation if the error is the hard type (feed check, machine check, etc).

Figure 5-23. 2560 Status Bytes

IBM 5424 Multi-Function Card Unit (MFCU)

The MFCU is a 96-column card input/output unit. Cards fed from either of two hoppers are read, punched, printed, and stacked in any of four stackers.

Figure 5-24 shows the path cards take through the MFCU. Two hoppers are provided: the primary and the secondary. Cards can enter the unit and be read from either hopper. After the reading station, cards from the primary go to an upper level wait station; cards from the secondary go to a lower level wait station. From these wait stations either the primary or the secondary card can be advanced through the punching and printing stations to the stackers.

The following combinations of operations can be initiated by a start I/O instruction:

- Feed
- Feed and read
- Punch and feed
- Punch, feed, and read
- Print and feed
- Print, feed, and read
- Punch, print, and feed
- Punch, print, feed, and read
- Selection of the card leaving the wait station into any of four stackers

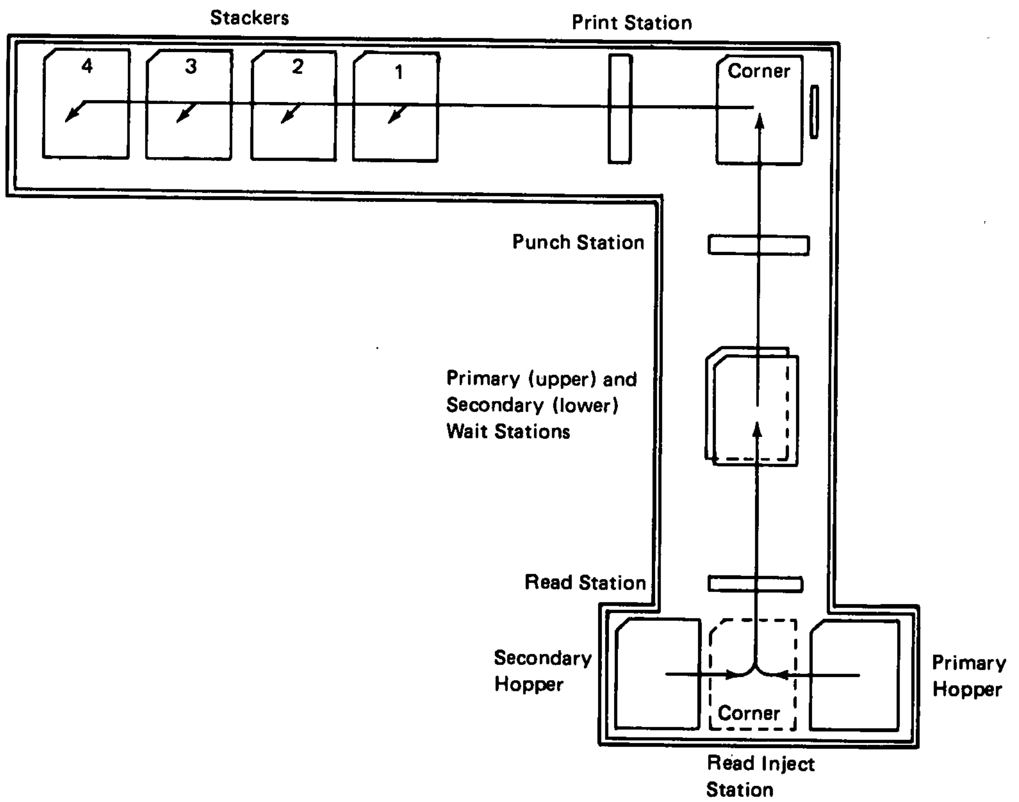


Figure 5-24. 5424 Card Path

5424 NOT-READY-TO-READY INTERRUPT—MODEL 15 ONLY

If interrupt level 6 is enabled, the 5424 sends an interrupt request to the system whenever it goes from a not-ready state to a ready state.

5424 OPERATIONS

5424 Card Read Operations

The read/feed functions of start I/O instructions move a card from the specified hopper to the corresponding wait station. If read is specified, the data contained in all 96 columns of the card is transferred to storage at a field specified by a load I/O instruction. The read data is checked to ensure that it is read correctly. An error in reading causes a read check.

A load I/O instruction must be executed before each start I/O instruction that specifies card reading. This load I/O instruction must load the address of the high-order byte of the read data field into the MFCU read data address register. To meet performance specifications the addresses must be on 128-byte boundaries.

The card feeding and reading rate is determined by the operations being performed. The rated reading speeds (250 cards per minute for Model A1 and 500 cards per minute for Model A2) are for read operations only. If punching or printing is performed at the same time, the reading rate will be reduced to the rate at which punching and printing are performed. To maintain the rated reading rate, successive start I/O instructions specifying reading must be issued within 44 milliseconds (Model A1) or 22 milliseconds (Model A2) after the read/feed busy indicator indicates not busy. The read/feed busy indicator can be tested with a test-I/O-and-branch instruction.

Program Notes: There are three MFCU print busy indicators. The card printer busy (testable with the TIO or APL instruction) comes on with the SIO instruction including print, and goes off with the start of printing on the actual card. For maximum hardware overlap for rated throughput, the next SIO instruction including print can be issued and will be accepted by the hardware at this time. Because printing for the first card has not been completed, error checking (for print errors) cannot be done at this time. When the next APL or TIO instruction is issued (after the second SIO), it will indicate any errors on the first card, since the first card is now complete and the second card has arrived at the print station. However, printing may have been started or even completed on the second card. Therefore, an error indicated at this time may have occurred on either of the two cards.

Print buffer 1 busy and Print buffer 2 busy (testable by SNS and TBN or TBF instructions) can be used to determine which MFCU print buffer (or buffers) is available. However, this busy indication drops just prior to the completion of the print operation. Consequently, an error condition can come up after this indication drops.

After the last I/O operation in a program, a final wait operation should be performed in which a wait is done on the card in transport/counter bits to become 0. This is to ensure that all cards have cleared the transport without feed checks and that no errors have occurred during the last I/O operations.

5424 IPL Read

Pressing the program load key causes the following reader actions to occur:

1. The MFCU read data address register is set to 0000.
2. A read operation is performed from the primary hopper of the MFCU without a start I/O instruction being executed.

The read operation is performed in the IPL card reading mode described in the introductory chapter of this manual. Reading in this mode (C and D bits taken from tier 3) can be continued by setting bit 1 of the start I/O instruction control code to 1 for each read start I/O instruction in which IPL mode reading is desired. IPL read can also be initiated by a start I/O operation.

5424 Punch Operations

Start I/O instructions that specify punching initiate moving a card from one of the wait stations, through the punch station and transport, to the stackers. As the cards pass through the punch station, data from storage is recorded in them in the form of punched holes. The punching is checked to ensure that the correct data is punched. An error causes a punch check. The punch data is checked to ensure that the data to be punched is valid for the 64 characters allowed in the card code. An error causes a punch invalid check. No punch checking is performed after a punch invalid check.

A load I/O instruction must be executed before each start I/O instruction that specifies a punch operation. This load I/O instruction places the address of the high-order byte of the punch data field in the MFCU punch data address register. Column 1 of the card is punched with the data contained in storage at this address. Column 2 of the card is punched with the data contained in storage at the next higher address. The punch data fields must be on 128-byte boundaries.

If a punch start I/O instruction is given with no card in the wait station, the instruction will be ignored and the no-op status bit will be set.

Card punching is performed at a single rate for each model of MFCU, Model A1 at 60 cards per minute and Model A2 at 120 cards per minute. To maintain this throughput, successive punch start I/O instructions must be executed within 90 milliseconds (A1) or 45 milliseconds (A2) of the end of punch busy indication to the test-I/O-and-branch instruction.

5424 Print Operations

The start I/O print and feed or print and read operation initiates card motion from the selected wait station, through the punch and cornering stations, and into the print station where three or four lines of 32 characters each are printed on the card. If there is no card in the wait station, the instruction is ignored and the no-op bit is turned on in the status indicators.

The print data area must be loaded before the start I/O print instruction is issued. The print data area consists of two print buffers each of which is always 128 bytes in length even though only 96 bytes are required when three lines are printed. The buffers are located in main storage. They are defined to the MFCU attachment with a load I/O instruction that loads the address of the high-order byte of print buffer 1 into the MFCU print data address register. The print data buffer address must be on a 256-byte boundary.

The load I/O instruction should be given only once for each job or each time the print data address area changes. If the load I/O instruction is given while either print buffer is busy, an unconditional program advance (or loop on the load I/O instruction) occurs until both buffers are free. This causes a loss of throughput. If power is lost for any reason, the print load I/O instruction must be re-executed before a start I/O instruction specifying printing is executed, or processor checks will occur if printing is attempted.

The 128-byte print data area is printed on the card in the following manner:

Line 1 — Leftmost address to byte 32
Line 2 — Bytes 33 through 64
Line 3 — Bytes 65 through 96
Line 4 — Bytes 97 through 128 if the fourth line of print is called for

The print buffer to be used is selected by setting bit 0 of the R-byte of the start I/O instruction to 0 for print buffer 1, and to 1 for print buffer 2.

The MFCU prints any of the 64 characters in the card code. Any of the characters in the 256-character EBCDIC set that is not included in the card code prints as a blank *without signaling the program.*

The rated throughput in print operations printing three lines is 60 cards per minute for the Model A1, 120 cards per minute for the Model A2. To maintain rated throughput, successive print operations must be initiated within 600 milliseconds (A1) or 300 milliseconds (A2) after the end of print data busy indication to a sense I/O and test bits operation.

5424 Combined Operation

Start I/O punch-print-read or punch-print-feed operations proceed in the same manner as described for individual operations except that one card is fed from the wait station, punched into, and printed on before stacking. The next card is fed from the specified hopper into the wait station during punching. If read was specified, the data in the card is read into storage. To maintain rated throughputs, successive punch, print operations must be initiated within 20 milliseconds after the end of the later of punch busy or print data busy indicators.

5424 Stacker Selection

Primary cards are selected to stacker 1 and secondary cards are selected to stacker 4 unless another stacker is specified. Stacker select is given by including the stacker select information in the start I/O control code of any of the start I/O instructions previously described. Stacker selection is performed on the card in the wait station when the start I/O instruction is executed, not the card that leaves the hopper. For programmed stacker select to operate, the stacker bit (bit 5) of the control code must be 1.

5424 START I/O (SIO)

Op Code (hex)	Q-Byte (binary)	R-Byte (binary)
Byte 1	Byte 2	Byte 3
F3	1111 x xxx	xxx0 0xxx

DA M N Control Code

Bits

0123 4567	Function Specified
xxx0 0100	Select stacker 4
xxx0 0101	Select stacker 1
xxx0 0110	Select stacker 2
xxx0 0111	Select stacker 3
xx10 0xxx	Print four lines
x1x0 0xxx	IPL read
0xx0 0xxx	Print buffer 1 (see *5424 Print Operations*)
1xx0 0xxx	Print buffer 2 (see *5424 Print Operations*)

N-Code	Operation
000	Feed
001	Feed and read
010	Punch and feed
011	Punch, feed, and read
100	Print and feed
101	Print, feed, and read
110	Punch, print, and feed
111	Punch, print, feed, and read

0 specifies operation on card or cards in primary feed.
1 specifies operation on card or cards in secondary feed.

Hex F specifies the 5424 as the device to be controlled.

F3 specifies a start I/O operation. F as the first hex character in the op code identifies a command-type instruction (that is, an instruction without operand addressing).

Operation

The processing unit tests the 5424 feed specified by the M-code for busy, not-ready, and check conditions. If none of these conditions exist, the 5424 performs the function or functions specified by the N-code and the R-byte.

If the specified feed is busy when the program issues the SIO, or if the 5424 is not ready for any reason except unit check, the program loops on the SIO instruction until the feed becomes not busy or the 5424 is made ready. Then the 5424 executes the instruction.

Program Notes

- When the SIO is issued while the MFCU is not ready, the processing unit lights the I/O ATTENTION light on the processing unit console to alert the operator that operator intervention is required.
- Whenever a 5424 feed check exists at the time the processing unit issues an instruction, the processing unit sets the no-op status bit (status byte 1, bit 7) and, if interrupts are enabled on Model 15, requests an op-end interrupt. Conditions causing no-ops are (1) feed check and (2) either a punch or print instruction being issued without a card in the wait station.
- Whenever a 5424 check that will not prevent the execution of the SIO instruction exists, the processing unit executes the instruction and resets that check bit.

5424 Op End Interrupt—Model 15 Only

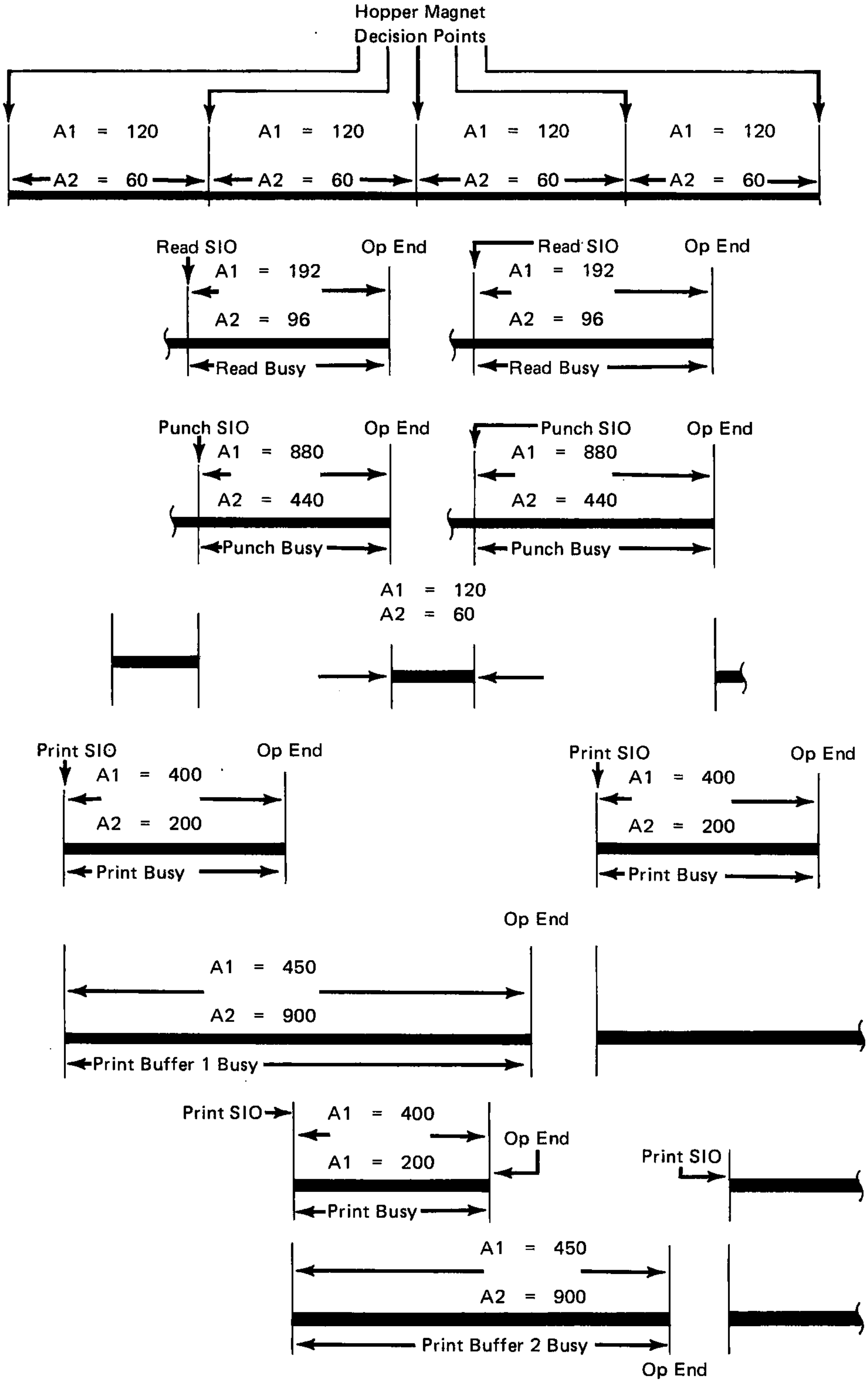
If interrupts are enabled, the 5424 attachment presents an op-end interrupt request to the processing unit at the end of the processing unit instruction in progress when one of the following conditions occurred:

1. The 5424 went from read busy to read not busy.
2. The 5424 went from punch busy to punch not busy.
3. The 5424 went from print busy to print not busy.
4. The 5424 print buffer 1 went from busy to not busy.
5. The 5424 print buffer 2 went from busy to not busy.
6. A 5424 feed check occurred.
7. A no-op condition was set.

See Figure 5-25 for 5424 Op-End timings.

5424 Interrupt Pending—Model 15 Only

The Model 15 program tests most attachments for an interrupt pending condition by means of a TIO instruction. However, 5424 interrupt pending conditions are tested by sense I/O instructions.



A1 = 5424 Model A1
A2 = 5424 Model A2
The numbers following A1 and A2 represent time in ms.
The 5424 initiates an op end interrupt request with each op end shown.

Figure 5-25. 5424 Op-End Interrupt Timings (Nominal Times)

5424 TEST I/O AND BRANCH (TIO)

Op Code (hex) Byte 1	Q-Byte (binary) Byte 2	Operand Address Byte 3	Byte 4
C1	1111 x xxx	Operand 1 address	
D1	1111 x xxx	Op 1 disp from XR1	
E1	1111 x xxx	Op 1 disp from XR2	

DA M N

N-Code	Condition Tested
000	Not ready/check
001	Read/feed busy
010	Punch data busy
011	Either read/feed or punch data busy (or both busy)
100	Card printer busy
101	Either read/feed or card printer busy (or both busy)
110	Either punch data or card printer busy (or both busy)
111	Any one, two, or all of the following: Read feed busy; Punch data busy; Card printer busy

M: 0 specifies the primary feed for testing.
1 specifies the secondary feed for testing.

DA: Hex F specifies the 5424 as the device being tested.

C1, D1, or E1 specifies a test I/O and branch operation. The first hex character on the op code specifies the type of operand addressing for the instruction.

Operation

The processing unit tests the 5424 primary or secondary feed (as specified by the M-code for any condition specified by the N-code (see *Q-Byte*). If one of the tested conditions exists, the program branches to the address in the operand portion of the instruction. If no tested condition exists, the program proceeds with the next sequential instruction.

Program Notes

- The address not used for the next sequential instruction (the branch-to address when a branch does not occur, or the next sequential instruction when a branch does occur) remains in the address recall register until the next decimal, insert-and-test-characters, or branch instruction is executed.
- Read/feed becomes busy as soon as a start I/O instruction for the MFCU is accepted by the MFCU. Punch data becomes busy when the MFCU accepts a start I/O instruction that specifies punching. Acceptance of an MFCU instruction that specifies printing causes a card printer busy indication. The card printer becoming not busy does not indicate that the print operation is complete, because this indication drops (to allow another print instruction to be issued) before the print operation is completed. The occurrence of a feed check while any one of the busy conditions is active turns off the busy condition immediately. Otherwise, the busy condition is turned off at the end of the I/O operation (except as noted for the card printer busy indication).

Resulting Condition Register Setting

This instruction does not affect the condition register.

5424 ADVANCE PROGRAM LEVEL (APL)

Op Code (hex) Byte 1	Q-Byte (binary) Byte 2	R-Byte (binary) Byte 3
F1	1111 x xxx	0000 0000

DA M N — R-byte is not used in an APL instruction

N-Code	Condition Tested
000	Not ready/check
001	Read/feed busy
010	Punch data busy
011	Either read/feed or punch data busy (or both busy)
100	Card printer busy
101	Either read feed or card printer busy (or both busy)
110	Either punch data or card printer busy (or both busy)
111	Any one, two, or all of the following: Read feed busy Punch data busy Card printer busy

M: 0 specifies the primary feed for testing
1 specifies the secondary feed for testing

DA: Hex F specifies the 5424 as the device being tested.

F1 specifies an advance program level operation. F as the first hex character in the op code specifies a command type instruction (that is, an instruction with no operand addressing).

Operation

This instruction tests for the conditions specified in the Q-byte.

- Condition present:
  - Systems with Dual Program Feature installed and enabled, activate the inactive program level.
  - Systems without Dual Program Feature installed or with Dual Program Feature installed but not enabled, loop on the advance program level instruction until the condition no longer exists.
- Condition not present: Systems with or without Dual Program Feature access the next sequential instruction in the active program level.

Program Notes

- Read/feed becomes busy as soon as a start I/O instruction for the MFCU is accepted by the MFCU. Punch data becomes busy when the MFCU accepts a start I/O instruction that specifies punching. Acceptance of an MFCU instruction that specifies printing causes a card printer busy indication. The card printer becoming not-busy does not indicate that the print operation is complete, because this indication drops (to allow another print instruction to be issued) before the print operation is completed. The occurrence of a feed check while any one of the busy conditions is active turns off the busy condition immediately. Otherwise, the busy condition is turned off at the end of the I/O operation (except as noted for the card printer busy indication).
- For additional information concerning the advance program level instruction, see Chapter 2.

5424 LOAD I/O (LIO)

Op Code (hex)	Q-Byte (binary)	Operand Address	
Byte 1	Byte 2	Byte 3	Byte 4
31	1111 x xxx	Operand 1 address	
71	1111 x xxx	Op 1 disp from XR1	
B1	1111 x xxx	Op 1 disp from XR2	

DA M N

N-Code	To Be Loaded
100	5424 print data address register
101	5424 read data address register
110	5424 punch data address register
111	*Models 10 and 12:* Invalid N-code; results in processor check *Model 15:* 5424 interrupt control register (The storage byte specified by the effective address holds the interrupt control code; the other byte is not used by the CPU. Allowed control bytes are shown below:

Control Code Bits[1] 0123	4567	Meaning
XYYY	1XXX	Enable interrupts
XYY1	YXXX	Reset op-end interrupt
XY1Y	YXXX	Reset print buffer 1 interrupt
X1YY	YXXX	Reset print buffer 2 interrupt
XYYY	0XXX	Disable interrupts

Any N-code not shown is invalid and causes:
- Program check if interrupt level 7 is enabled on Model 15
- Processor check if interrupt level 7 is not enabled on Model 15
- Processor check on Models 10 and 12

M: 0 specifies normal mode operations.
1 specifies diagnostic mode operations.

DA: Hex F specifies the 5424 as the addressed device.

31, 71, or B1 specifies a load I/O operation. The first hex character in the op code specifies the type of operand addressing to be used for the instruction.

---

[1] X = Should be 0, but may be 1
Y = Can be 1 if multiple interrupt control functions are desired; otherwise, must be 0

Operation

The processing unit loads the 2 bytes of data contained in the operand into the register specified by the N-code. If the selected register is busy, the program loops on the load I/O instruction until the register becomes not busy.

Program Note, General

The 2-byte operand is addressed by its higher numbered position.

Program Notes, Model 15

- All pending interrupts are reset and lost when interrupts are disabled.
- Interrupt requests will not occur as a result of operations that end when interrupts are disabled.
- Interrupt pending can be tested by sensing byte 2, bit 3 with a sense I/O instruction that has an N-code of 000.
- The interrupt request source can be tested by TIO and SNS instructions.

5424 SENSE I/O (SNS)

Op Code (hex)	Q-Byte (binary)	Operand Address	
Byte 1	Byte 2	Byte 3	Byte 4
30	1111 0 xxx	Operand 1 address	
70	1111 0 xxx	Op 1 disp from XR1	
B0	1111 0 xxx	Op 1 disp from XR2	

DA M N

N-Code	Sensed Unit
000	Byte 2, bit 3 = interrupt pending. All other bits are CE diagnostic bits.
001	CE diagnostic bytes
011	Status bytes
100	5424 print data address register
101	5424 read data address register
110	5424 punch data address register

Any N-code not shown is invalid and causes:
- Program check if interrupt level 7 is enabled on Model 15
- Processor check if interrupt level 7 is not enabled on Model 15
- Processor check on Models 10 and 12

M: Not used; should be 0

DA: Hex F specifies the 5424 as the addressed device.

30, 70, or B0 specifies a sense I/O operation. The first hex character in the op code specifies the type of operand addressing for the instruction.

Operation

The CPU transfers 2 bytes of data from the unit specified by the N-code to the main storage field specified by the operand address. The first byte transferred enters the effective address (the operand address), the second byte enters the effective address minus 1. Status bits are described in Figure 5-26.

Byte	Bit	Name	Indicates	Reset By
1	0	Read check	Data is read incorrectly.	SIO, system reset, NPRO, check reset
1	1	Punch check	Correct punches not selected by MFCU.	
1	2	Punch invalid	CPU sent MFCU nonpunch character for punching.	SNS specifying status indicators (N=011), system reset, check reset, NPRO
1	3	Print data check	Print wheel is out of synchronism.	
1	4	Print clutch check	Card was printed on wrong line (is too high or low).	
1	5	Hopper check	No card left hopper during execution of feed-type instruction.	NPRO, pressing MFCU START key
1	6	Feed check	Any incorrect card movement in card path.	NPRO
1	7	No-op	CPU issued command MFCU cannot execute.	SNS specifying status indicators (N=011), system reset, check reset, NPRO
2	0	Print buffer 1 busy	MFCU has accepted SIO specifying printing from buffer 1 (operand byte bit 0 = 0).	Printing operation for card being complete
2	1	Print buffer 2 busy	MFCU has accepted SIO specifying printing from buffer 2 (operand byte bit 0 = 1).	
2	2	Card in wait 1	Read/feed has become not busy following a read or feed operation that moved a card from the primary hopper.	
2	3	Card in wait 2	Read/feed has become not busy following a read or feed operation that moved a card from the secondary hopper.	
2	4	Overrun	This condition should not occur. It indicates that requests for cycle steals were not granted fast enough to handle each byte of I/O data. Overruns result in a loss of I/O data. IBM required special equipment engineering can determine whether configurations involving high data rate devices, such as the RPQ items installed, will result in data overrun if IBM program products are not being used. Contact your IBM sales representative for this information.	
2	5	Hopper cycle not complete	A start I/O command has been accepted for execution, but the card has not been moved completely from the hopper.	Card moving out of the hopper
2	6	Card in transport counter (binary) bit 2	These 2 bits constitute a counter that keeps track of the number of cards between the wait station and the stackers. Every card that leaves the wait station adds 1 to the counter. Every card that is directed to a stacker, except those stacked after a machine check, subtracts 1 from the counter. When a feed check occurs, the counter indicates the number of cards that were in the transport when the feed check occurred. These bits are reset to 0 by turning power on and by non-process runout.	NPRO and powering up
2	7	Card in transport counter (binary) bit 1		

Figure 5-26. 5424 Status Bytes

IBM 3410/3411 Magnetic Tape Subsystem

The 3410/3411 Models 1, 2, and 3 tape subsystems read and write half-inch magnetic tape. The IBM 3410 Magnetic Tape Unit is a tape unit only; the IBM 3411 Magnetic Tape Unit and Control is a tape unit and a control unit in the same frame.

A 3410/3411 magnetic tape subsystem is available in one of the following configurations for attachment to a System/3:

- One 3411 Model 1, 2, or 3
- One 3411 Model 1, 2, or 3 and one 3410[1]
- One 3411 Model 1, 2, or 3 and two 3410s[1]
- One 3411 Model 1, 2, or 3 and three 3410s[1]

3410/3411 PERFORMANCE SUMMARY

Figure 6-1 shows the performance information for the 3410/3411 tape subsystem.

[1] Same model number as the 3411.

	Model 1	Model 2	Model 3
Tape speed (in./sec)	12.5	25	50
Interblock gap (IBG)[1]: Length/time (9-track) Length/time (7-track)	 0.6 inch (48 ms) 0.75 inch (60 ms)	 0.6 inch (24 ms) 0.75 inch (30 ms)	 0.6 inch (12 ms) 0.75 inch (15 ms)
Write access time[2]	15 ms	12 ms	6 ms
Read access time[2]	15 ms	12 ms	6 ms
Data rate: 1600 bpi 800 bpi 556 bpi 200 bpi	 20K bytes/sec 10K bytes/sec 6.95K bytes/sec 2.5K bytes/sec	 40K bytes/sec 20K bytes/sec 13.9K bytes/sec 5.0K bytes/sec	 80K bytes/sec 40K bytes/sec 27.8K bytes/sec 10K bytes/sec
Time per byte: 1600 bpi 800 bpi 556 bpi 200 bpi	 50 μs 100 μs 144 μs 400 μs	 25 μs 50 μs 72 μs 200 μs	 12.5 μs 25 μs 36 μs 100 μs
Rewind time (±10%)	3 min/2400 ft	3 min/2400 ft	2 min/2400 ft
Reel sizes (inch)	10.5, 8.5, 7, 6	10.5, 8.5, 7, 6	10.5, 8.5, 7, 6
Tape threading	Manual	Manual	Manual
Tape motion	Tape is driven by a single capstan that is directly coupled to a low-inertia, high-torque, dc motor.		
Read/write head	The chrome-plated, two-gap head is located in the left vacuum column.		

[1]An interblock gap is erased tape which separates blocks of data.
[2]Time given is for a 0.6 inch interblock gap.

Metric Equivalents:

1600 bpi	= 63 bytes per mm	0.6 inch	= 15.2 mm
800 bpi	= 31.5 bytes per mm	0.75 inch	= 19 mm
556 bpi	= 21.9 bytes per mm	6 inches	= 152.4 mm
200 bpi	= 7.9 bytes per mm	7 inches	= 177.8 mm
50 in./sec	= 1270 mm per second	8.5 inches	= 216 mm
25 in./sec	= 635 mm per second	10.5 inches	= 266.7 mm
12.5 in./sec	= 317.5 mm per second	2400 feet	= 732 meters

Figure 6-1. Performance Information for the 3410/3411 Tape Subsystem

3410/3411 SPECIAL FEATURES

Each 3410 and 3411 tape unit must be equipped with a special feature that specifies the read/write format desired. The features are:

Single density
Dual density
Seven-track

Any tape unit in the subsystem (a 3411 or an attached 3410) can be equipped with either the dual density feature or the seven-track feature, but not both.

A subsystem can have the following combination of features:

- Each tape unit has the single density feature.
- Each tape unit has the dual density feature.
- Each tape unit has the seven-track feature.
- Some tape units have the single density feature; some have the dual density feature.
- Some tape units have the single density feature; some have the seven-track feature.

Single Density Tape Unit Feature

This feature is installed on tape units to enable nine-track, phase-encoded (PE) operations. Single density control is standard on the 3411.

Dual Density Tape Unit Feature

This feature is installed on tape units to enable nine-track operations in both 1600 bpi phase-encoded (PE) mode and 800 bpi non-return-to-zero (NRZI) mode. If any tape unit is equipped with the dual density feature, the 3411 must also be equipped with the dual density control feature.

Seven-Track Tape Unit Feature

This feature can be installed on the tape unit portion of the 3411 or on any 3410. It enables the tape unit to read and write data in NRZI mode on seven-track magnetic tape. Reading and writing are done at densities of 200, 556, or 800 bpi. Odd or even parity is provided.

If the seven-track tape unit feature is installed in the 3411 or any attached 3410, the 3411 must also be equipped with a seven-track tape control feature. A 3410 or 3411 tape unit equipped with the seven-track tape unit feature cannot be equipped with either a single density tape unit feature or a dual density tape unit feature.

Dual Density Control Feature

This feature, available for the 3411 control unit, enables the tape units to read and write nine-track tape in either the 800 bpi NRZI or 1600 bpi PE mode.

The program must issue a mode set command to the tape control unit to set the desired writing density. A read operation does not require a mode set operation. When reading, a burst of bits in the parity track at load point identifies, to the tape unit, tape written at 1600 bpi. The lack of this burst identifies tape written at 800 bpi.

Seven-Track Control Feature

This feature, available for the 3411 control unit, enables the 3411 to control any tape unit (including the tape unit portion of the 3411) that is equipped with the seven-track tape unit feature.

A translator and data converter are included with the seven-track control feature. The translator, when set on, translates 8-bit bytes from main storage to 6-bit BCD tape characters and vice versa. Each main storage byte becomes a tape character; each tape character becomes 1 byte in main storage. The data rate is not changed by the translator.

The data converter allows writing and reading of binary data on seven-track tape units. Writing a tape with the data converter on causes four tape characters (24 bits) to be written for every 3 storage bytes (24 bits). Reading such a tape reverses the process by converting four tape characters into 3 storage bytes. Data conversion reduces the data transfer rate by 25 percent of that for nine-track NRZI operations. An odd/even count is made during read/write data converter operations to ensure correct transfer of data. An unequal count sets the data converter check bit.

The mode 1 set command bits (Figure 6-2) turn the translator and data converter on and off. The translator and data converter cannot be on at the same time, and the data converter cannot be used with a read backward command.

3410/3411 FUNCTIONAL CHARACTERISTICS

3410/3411 Tape Unit Control

The 3411 houses the clocks, delays, and controls necessary to operate the tape units attached to the system. These circuits receive instructions from the system through a tape attachment feature in the 5410. Upon receipt of the instructions, the control circuits, using a microprogram, direct all required tape unit operations and signal the system when the task is complete.

3410/3411 Operator Controls

Each tape unit has an operator panel that contains all subsystem manual controls. The 3411 also has an enable/disable switch on the operator panel to switch the subsystem online and offline.

3410/3411 File Protection

The 3410/3411 subsystem uses a plastic write-enable ring mounted on the tape reel to permit writing. If a tape is mounted without the ring in position, writing cannot occur; therefore, the file is protected.

3410/3411 Tape Requirements

The following half-inch tapes can be used: IBM Series/500, IBM Heavy Duty, IBM Dynexcel, or competitive formulations which meet the tape and reel criteria in *Tape Specifications*, GA31-0006.

Note: IBM tapes other than those named above do not provide adequate reliability and should not be used.

3410/3411 Erase Head

The erase head applies a strong magnetic field that erases the entire tape width during write or erase operations. Full-width erasure eliminates extraneous bits in interblock gaps or skip areas, and destroys previously written bits.

3410/3411 Parity Checking

During write operations, each byte is parity checked twice: when it is received from the system and when it is written on tape (read back checking). During read operations, each byte is parity checked before it is sent to the system, and single-track errors are corrected. During sense operations, the tape control supplies proper parity for each byte. The tape control parity checks all bytes received from the system.

3410/3411 Tape Subsystem Servicing

The tape subsystem is attached to the system in such a manner that it usually can be serviced offline without impacting other system operations.

3410/3411 Cabling

The 3411 is connected by cable to the system through an opening in the IBM 5203 or 5421; the first attached 3410 is internally attached to the 3411, the second 3410 is internally attached to the first 3410, and the third 3410 is internally attached to the second 3410.

3410/3411 Addressing

Each tape unit has a fixed address.

3410/3411 Intersystem Tape Exchange

Tapes produced on the 3410/3411 subsystem and all other IBM half-inch tape units operating in the same density are interchangeable. Therefore, output data produced on one system, such as the IBM System/360 or System/370, can be used as direct input to another system, such as System/3 using compatible tape.

3410/3411 TAPE UNIT OPERATIONS

The processing unit initiates an I/O operation on a tape unit with the start I/O instruction. Figure 6-2 shows the bit settings for read, read backward, and write operations.

Control Code Formats

0	1	2	3	4	5	6	7	
0	0	C	C	C	1	1	1	Tape motion
M	M	M	M	M	0	1	1	Mode 1 set (seven-track)
1	1	0	0	D	0	1	1	Mode 2 set (nine-track)
1	0	0	1	0	1	1	1	Data security erase
0	0	0	0	0	1	0	1	Loop-write-to-read

C C C	(Control Code)
0 0 0	Rewind
0 0 1	Rewind/unload
0 1 0	Erase gap
0 1 1	Write tape mark
1 0 0	Backspace block
1 0 1	Backspace file
1 1 0	Forward space block
1 1 1	Forward space file

D	(Mode 2 Set)
0	1600 bpi
1	800 bpi

M M M M M	Density (bpi)			Parity		Data Converter On	Data Converter Off	Translator On	Translator Off
(Mode 1 Modifiers)	200	556	800	Odd	Even				
0 0 0 1 0	X			X		X			X
0 0 1 0 0	X				X		X		X
0 0 1 0 1	X				X		X	X	
0 0 1 1 0	X			X			X		X
0 0 1 1 1	X			X			X	X	
0 1 0 1 0		X		X		X			X
0 1 1 0 0		X			X		X		X
0 1 1 0 1		X			X		X	X	
0 1 1 1 0		X		X			X		X
0 1 1 1 1		X		X			X	X	
1 0 0 1 0			X	X		X			X
1 0 1 0 0			X		X		X		X
1 0 1 0 1			X		X		X	X	
1 0 1 1 0			X	X			X		X
1 0 1 1 1			X	X			X	X	

Figure 6-2. 3410/3411 Tape Command Code Formats

3410/3411 Read

A read forward operation is defined by the Q-byte. The tape unit moves tape forward, assembling the data from tape. Whenever a byte of data is available, a data transfer cycle is requested until the byte count reaches 0. The magnetic tape address register (MTAR) is increased by one after each data transfer cycle (more information about this is given in the note under *3410/3411 Load I/O (LIO)* in this section). The data is placed in contiguous ascending locations in main storage.

The unit exception condition is set if a tape mark is detected. The EOT (end-of-tape) reflective marker is not recognized during read forward operations.

Note: Seven-track tapes read in the incorrect mode or on nine-track units can result in data checks or tape runaway conditions.

3410/3411 Read Backward

A read backward operation is defined by the Q-byte. The tape unit moves tape backward and places data in storage in reverse of the order in which it was written. The MTAR is decreased by 1 after each data transfer cycle. The unit exception condition is set if a tape mark is detected.

Note: Excessive error indications can result if a seven-track read backward operation is attempted using tapes generated by IBM tape models, or others, prior to the IBM 2400 series tape units.

3410/3411 Write

A write operation is defined by the Q-byte. The tape unit moves tape forward, writing data from main storage. The subsystem remains busy until after read back checking of the written data.

The EOT reflective marker indicates that about 25 feet of tape remains on the reel. Ignoring this indication can unwind tape off the reel. When repositioning tape past the EOT marker, the only indication guaranteed is when the reflective marker is first sensed.

Note: The recommended minimum block length is 18 bytes.

3410/3411 Control

A tape control command is performed when the SIO Q-byte, bits 5, 6, and 7, are 0. Then the R-byte defines the control command. Figure 6-2 shows the command code bit settings. An interrupt control command is performed when the LIO N-code = 110. A control command is initiated at the tape control and tape unit. No transfer of data is involved.

3410/3411 Tape Motion Control Commands

Rewind: This command rewinds the tape. The tape unit remains loaded when the tape reaches load point. The tape unit is busy until the tape unit reaches the load point.

Rewind/Unload: This command rewinds the tape to load point and automatically unloads it. If the tape unit is at load point when the command is issued, tape immediately unloads because no rewind is required. The tape unit becomes not-ready after accepting a rewind/unload command. It remains not-ready until made ready by the operator. The subsystem is busy only until the tape unit accepts and begins executing the command.

Erase Gap: The tape unit moves forward, erasing tape for a distance of approximately 3-1/2 inches. When this operation is performed in the end-of-tape area, it sets the unit exception condition. The subsystem is busy during an erase gap operation.

Write Tape Mark: This command causes the subsystem to write a tape mark. A tape mark is a block of special non-data bytes separating files. Tape mark formats are predetermined in the subsystem and require no communication with the system while writing the tape mark. When this operation is performed in the end-of-tape area, it sets the unit exception condition. The subsystem is busy while the tape mark is being written.

Forward Space Block: This command moves tape forward to the next interblock gap. Data is not transferred, and errors associated with that block of data are not detected. When a tape mark is sensed, it sets the unit exception condition. The EOT reflective marker is not recognized during this operation. The subsystem is busy during a forward space block operation.

Backspace Block: This command moves tape backward to the nearest interblock gap or to load point, whichever comes first. No data is transferred. When a tape mark is sensed, it sets the unit exception condition.

Forward Space File: This command moves tape forward to the interblock gap beyond the next tape mark. Data is not transferred and data errors are not detected. The EOT reflective marker is not recognized during this operation. The subsystem remains busy until a tape mark is detected.

Backspace File: This command moves tape backward to the interblock gap beyond the next tape mark or to load point, whichever comes first. Data is not transferred and data errors are not detected. The subsystem remains busy until a tape mark or the load point is detected. If the load point marker is detected, the not ready/unit check and backward-at-load-point conditions are set.

Data Security Erase: This command erases tape from the tape's present position to the EOT marker. The subsystem remains busy until the tape unit accepts and begins executing the command. The tape unit remains busy until the erase is completed. This command must be issued only after the tape unit is put in write status. If the tape unit is in read status or is file protected when a data security erase command is issued, the command reject and not read/unit check conditions are set.

Erasing data beyond the EOT marker is the responsibility of the user. Fifteen erase gap commands erase about 4-1/2 feet of tape.

This command is accepted and terminated without error if it is issued when the tape is at end of tape and the tape unit is in write status. However, if the tape unit is in read status, a command reject condition is set. The subsystem does not present busy status in either case.

Loop-Write-to-Read: This command is used for diagnostic purposes. It is defined by the Q-byte setting.

3410/3411 Mode Set Commands

Mode set commands are used to select density, parity, data converter, and code translator for seven-track operations. Figure 6-2 shows the mode modifier bit settings that set these conditions. Figure 6-3 gives the subsystem response to mode set commands for write operations.

Mode 1 Set: This sets the control unit to the seven-track NRZI operation. It affects operation of all seven-track tape units attached to the tape control. Unless reset, the tape control retains its mode setting until it receives another mode 1 set command. A system reset forces a default condition of hex 93.

Mode 2 Set: This sets dual density tape controls to either 1600 bpi (PE) or 800 bpi (NRZI) mode for succeeding write operations. It is effective only when tape is positioned at load point and the tape unit is in ready status. The tape unit retains this mode setting until the tape again reaches load point, at which time the tape unit is automatically set to PE mode (this also applies to rewind operations). The control unit retains the last mode set, and successive operations are performed in that mode unless there is a system reset. The tape control is set to PE mode after a reset.

Feature Installed on Control Portion of 3411 and Selected 3410/3411 Tape Units	Action Taken by Subsystem (Write) 1600 bpi	800 bpi	800/556/ 200 bpi
Dual density control feature and dual density tape unit feature:			
1600 bpi mode set[1]	X		
800 bpi mode set[1]		X	
Seven-track mode set	Previous setting		
No mode set	Previous setting		
Dual density control feature installed/uninstalled and dual density tape unit feature not installed:			
1600 bpi mode set	X		
800 bpi mode set	X		
Seven-track mode set	X		
No mode set	X		
Seven-track control feature installed and seven-track tape unit feature installed:			
1600 bpi mode set			Previous setting
800 bpi mode set			Previous setting
Seven-track mode set			Density specified
No mode set			Previous setting
Seven-track control feature installed and seven-track tape unit feature not installed:			
1600 bpi mode set	X		
800 bpi mode set	X		
Seven-track mode set	X		
No mode set	X		

[1]Effective only at load point; if at other than load point, the previous setting is used.

Figure 6-3. 3410/3411 Subsystem Response to Mode Set Commands for Write Operations

SUGGESTED 3410/3411 ERROR RECOVERY PROCEDURES

The following minimum error recovery procedures are defined for the 3410/3411 tape subsystem to achieve acceptable performance and read/write reliability.

General Actions, 3410/3411

The system logs the number, severity, and type of I/O errors that occur while processing each reel of tape. This log helps the CE determine whether a problem is tape or machine oriented.

An operating system allows:

- Operator control
- Additional programmed recovery

Some of the operator control options that can be defined are:

- Retry the recovery procedure
- Continue to additional programmed recovery
- Dump the failing record
- Cancel the job

3410/3411 Messages

Any operator message (printed or message display unit) issued for error recovery procedures should contain the following information:

- Message code
- Error condition that caused the message

3410/3411 Sense Procedures

When an error occurs, sense information must be taken as follows:

1. Obtain and analyze attachment sense bytes 0 and 1 before attempting any subsystem sense byte actions.
2. Perform attachment sense byte actions.
3. Obtain and analyze attachment sense bytes 0 and 1 valid sense.
4. Obtain sense bytes 2-3, 4-5, and 6-7 (in that order) when the sense valid bit is set.

When sense is (or has become) valid, successive sense instructions must be executed within 30 ms of each other. Otherwise, the sense information in bytes 2-7 becomes invalid due to normal subsystem activity.

3410/3411 Sense Instructions

Attachment sense (2 bytes) and subsystem sense (3 bytes) must be executed to the failing unit, without any intervening instructions to the subsystem, when an error is detected. The subsystem hardware error sense byte (an additional sense byte) is available for hardware error information. This byte is sensed only when specified by the error recovery procedure. Update the tape error statistics with this sense information.

The sense bytes and bits must be tested in the order (priority) shown in Figure 6-4, and the actions must be performed as described in Figure 6-5.

Priority	Byte	Bit	Condition	Applicable for			Perform Action[1]
				Read	Write	Control	
Attachment Sense							
1	0	3	Tape control disabled	X	X	X	A
2	0	5	Subsystem busy	X	X	X	B
3	0	1	ABI parity error	X	X	X	C
4	0	2	ABO parity error	X	X	X	C
5	0	4	Two tag error	X	X	X	D
6	0	6	Sequence error	X	X	X	D
7	–	–	No error found	–	–	–	E
Subsystem Sense							
8	0	7	Sense valid	X	X	X	I
9	0	6	Equipment check	X	X	X	II
10	5	6	PE ID burst check		X	X	IV
11	0	5	No-op	X	X	X	III
12	0	0	Noise	X		X	V
13	0	3	Data check	X			VI
13	0	3	Data check		X		VII
13	0	3	Data check			X	VIII
14	0	2	Unit exception	X	X	X	IX
15	0	1	Wrong length record	X			X
16	–	–	No error indicated	X	X	X	XI

[1] Described in Figure 5-5

Figure 6-4. 3410/3411 Attachment and Subsystem Sense Information

Action	
A	1. If the subsystem is busy, issue a message to tell the operator to enable the tape unit, then stop. Upon operator restart, proceed with the job. 2. If the subsystem is not busy, perform a subsystem hardware error sense operation, issue an operator message, and stop. Attempt at least one job restart if a hardware error occurred.
B	1. Repeat the attachment sense operation at least 15 times. 2. If the busy condition persists, perform a subsystem hardware error sense operation. Log the error, issue an operator message, and stop. 3. If the busy condition ends, continue checking the sense information.
C	Log the error and retry the operation up to 15 times. If unsuccessful, the condition becomes a permanent error.
D	1. Perform a subsystem hardware error sense operation. 2. Log the error, issue an operator message, and await operator action. 3. One job restart is recommended. (These errors are not recoverable and can be reset only with a system reset.)
E	Continue checking the subsystem sense bytes.
I	1. This bit on indicates a valid sense; continue checking the subsystem sense bits. 2. This bit off indicates an invalid sense; repeat the entire sense operation.
II	1. Perform a complete sense operation and make the information available to the CE. 2. Total all current statistical counters and make the total available to the CE. 3. Issue an operator message and await operator action. 4. One job restart is recommended.

Action	
III	This condition is set by subsystem sense byte 1. 1. If bit 1 is on, issue a message to tell the operator to install the write-enable ring and await operator action. Reissue the command sequence. 2. If bit 2 is on, the error can be handled by the operating program. If so, return to the operating program; otherwise, log the error, issue an operator message, and await operator action. 3. If bits 3 or 5 are on, reissue the instruction sequence up to 15 times. If the error persists, issue an operator message and await operator action. 4. If bits 4, 6, or 7 are on, log the error, issue an operator message, and await operator action.
IV	1. Rewind tape and reissue the command sequence up to 15 times. 2. If the error is correctable, log the error and return to the operating program. 3. If the error is not correctable in 15 retries, log the error, issue an operator message, and await operator action. *Suggested operator action:* Move the load point marker 1 or 2 centimeters toward the center of the tape reel and restart.
V	1. *Read operation:* Tell the operating program about the condition and let it decide whether this is a noise block on tape. If this condition is treated as a noise block, log the error, and return to the operating program. If a block of 12 or fewer bytes was expected, retry the read as outlined in *Action VI.* If a block of 12 or fewer bytes was not expected, discard the data as a noise block and reissue the same instruction sequence to read the expected block. 2. *Erase gap operation:* Perform *Action VII.* If successful, tell the operator an erase gap check occurred. If unsuccessful, a permanent write error results. 3. This condition may also result from generating a tape that cannot be read properly. There is no reliable error recovery for this failure except a job restart. The error may be due to faulty tape or a temporary erase error. Give the operator the option of proceeding with the job, after the message is given, or cancelling the job.

Figure 6-5 (Part 1 of 2). 3410/3411 Tape Error Recovery Procedures

Action	
VI	The data converter is invalid during read backward operations in seven-track mode. A mode 1 set command can be issued at any time, whether required or not. Retry the read 40 times in the same direction as the original read, then 40 times in the opposite direction. 1. Space the block in the opposite direction of the read in which the error occurred. 2. Set correct seven-track mode (if seven-track tape) and reissue the instruction sequence to reread in the same direction in which the error was originally detected. If the error does not exist, proceed to step 8. If the error remains, repeat steps 1 and 2 three more times. If the error still persists after a total of four rereads, issue a cleaner-blade operation as described in step 7, and return to step 3. 3. Read in the direction opposite that performed in step 2. If the error does not exist, proceed to step 8. If the error persists, proceed to step 4. 4. Space block in the direction opposite that performed in step 3. 5. Set correct seven-track mode and reissue the instruction sequence to read in the same direction as in step 3. 6. If the error persists, repeat steps 4 and 5 three more times. If the error still persists after a total of four rereads, issue a cleaner-blade operation as described in step 7. If this corrects the error, proceed to step 8. If the error persists, repeat steps 1 through 6 ten times. If the error still persists after all this, proceed to step 9. 7. *Cleaner-blade operation:* Perform this operation by issuing five backspace block commands (followed by five forward space block commands. If, during a tape cleaner operation, load point is reached in *n* backspaces, reposition the tape with *n*-1 forward spaces. If a tape mark is encountered in *n* space block operations, reposition the tape with *n* space block commands in the opposite direction. Repeat steps 1 through 6 until the record is read successfully or until a minimum of 80 retries (as described in steps 1 through 6) have been performed. 8. Log the error and return to the operating program. 9. The error is permanent if it still persists. Perform a complete sense I/O by issuing a loop-write-to-read operation using any available data, then testing for not ready/unit check. If the condition is satisfied, perform a complete sense operation and log the sense information. If the condition is not met, log the previous sense information. In either case, make the total of all statistical counters available to the CE, then issue an operator message and await operator action.

Action	
VII	1. Check for unit exception. If unit exception (end of tape) is not properly handled, it can be lost; and writing off the end of the tape can result. 2. Reposition the tape, issue an erase gap command, and reissue the instruction sequence. Repeat the procedure 15 times. If the error does not recur during the retry, log the error and return to the operating program. If the error persists, follow the procedures in *Action VI,* step 9, but use the previous data for the loop-write-to-read operation.
VIII	1. *Erase gap command:* This command is performed by issuing a rewind command and reissuing the original command. Repeat the procedure at least 15 times. If the error does not recur during the retry, log the error and return to the operating program. If the error persists, follow the procedure described in *Action VI,* step 9. 2. *Commands other than erase gap:* Log the error, issue an operator message, and await operator action.
IX	1. Log a unit exception condition. 2. Return to the operating program.
X	1. Log a wrong length record condition. 2. Return to the operating program.
XI	This condition indicates an unexpected tape unit status. If none of the following conditions exist, log the error and await operator action. One job restart is recommended. If any of the following conditions exist, log the error, issue an operator message, and await operator action: a. Subsystem sense byte 2, bit 6 indicates that the tape unit had a failure. Subsystem sense byte 6, bits 0-3 define the failure. b. Subsystem sense byte 2, bits 5 and 7 off, indicates a power-off condition on the tape unit or a disconnection from the tape unit. c. Subsystem sense byte 2, bits 5 and 7 on, indicates a tape unit not ready. Expect one of these conditions: dropped ready, manually reset once it was in ready status, or a rewind/unload was issued.

Figure 6-5 (Part 2 of 2). 3410/3411 Tape Error Recovery Procedures

3410/3411 ERROR RECORDING AND ERROR STATISTIC COUNTER ASSIGNMENTS

Figure 6-6 shows the format of the combined volume error and statistical data recording counters. These counters should be updated immediately before stopping because of an uncorrectable error and also before returning to the operating program when an error has been corrected. The attachment and subsystem sense bytes, logged because of an error, should be printed out daily or after each job.

Counter	Bytes	By Unit	By Volume
Noise blocks	1	X	X
Write skips	1	X	X
Start I/O	2	X	X
Temporary read forward	1	X	X
Temporary read backward	1	X	X
Temporary write	1	X	X
Diagnostic track error	1	X	
Short gap mode	1	X	
Multitrack error	2	X	
End data check	1	X	
Envelope check	1	X	
End velocity	1	X	
TIE byte	4	X	
Volume identification	1	X	
Device type	1/2	X	
Overrun	1	X	
Tape mark check	1	X	
PE ID burst error	1	X	
Start velocity	1	X	
Write feedthrough	1	X	
False end	1	X	
No readback data	1	X	
VRC	1	X	
First and last volume serial number	6	X	
Q-byte	1	X	
Tape density	1/2	X	
Block length	2	X	

Figure 6-6. 3410/3411 Combined Volume Error and Statistical Data Recording Counters

3410/3411 START I/O (SIO)

Op Code (hex) Byte 1	Q-Byte (binary) Byte 2	R-Byte (binary) Byte 3
F3	011x x xxx	xxxx xxxx

Q-Byte: bits DA (device address), M, N; R-Byte: Control Code.

N	Control Code Bits 0123 4567	Operation
000	1100 0011	Mode set (9 track PE)
	1100 1011	Mode set (9 track NRZI)
	0000 0111	Rewind
	0000 1111	Rewind unload
	0001 0111	Erase gap
	0001 1111	Write tape mark
	0010 0111	Backspace block
	0010 1111	Backsapce file
	0011 0111	Forward space block
	0011 1111	Forward space file
	1001 0111	Data security erase
001		Read forward
010		Write
011		Read backward
100	0000 0001	Diagnostic write
	0000 0011	Loop write to read
	0000 0111	Load byte
	0000 1000	Write skew check
	0000 1001	Read forward skew check
	0000 1101	Read backward skew check
101	0000 0010	Crosstalk check
	0000 0100	FWD diagnostic measure
	0000 0110	IBG timing test
	0000 1100	BKWD diagnostic measure
110		Attachment write diag
111		Attachment read diag

Bits 01234	Tape Unit Specified
01100	0
01101	1
01110	2
01111	3

F3 specifies a start I/O operation. F as the first hex character in the op code specifies a command-type instruction (that is, an instruction without operand addressing).

Operation

The tape unit specified by the DA and M-codes performs the function specified by the control code if N=000 or 1xx. The specified tape unit performs the function specified by the N-code only if N=001, 010, or 011.

Program Notes

- The program must specify a starting address and a record length prior to each read, read backward, or write operation. Otherwise, the attachment sets the not-ready/unit check and no-op bits and requests an op-end interrupt.
- The I/O working condition becomes active when the attachment accepts the instruction.
- The instruction is not executed if the no-op bit is on, the I/O attention or busy condition exists, or Q-byte parity is incorrect.
- Any start I/O instruction resets all sense information except no-op, which is reset only by sensing sense byte 0.

3410/3411 Op-End Interrupts—Model 15

The 3410/3411 operate on interrupt level 5. The attachment has five subinterrupt levels: one for the attachment itself and one for each of the four tape units. The attachment presents op-end interrupt request to the CPU at the end of the CPU instruction during which one of the following occurs:

- The attachment drops out of I/O working status.
- One of the tape units goes from busy to not-busy.

The program, upon receiving an op-end interrupt request, can test the 3410/3411 attachment with a TIO instruction to determine if the tape subsystem requested the interrupt. If so, sensing byte 0 indicates which unit needs programming action: the attachment, one of the tape units, or combinations of these units. After the CPU services the interrupt, the program issues an LIO interrupt control instruction to the unit just serviced to reset the subinterrupt level.

All tape unit subinterrupt levels are enabled or disabled by a single LIO instruction, although they can be reset separately.

3410/3411 LOAD I/O (LIO)

Op Code (hex)	Q-Byte (binary)	Operand Address	
Byte 1	Byte 2	Byte 3	Byte 4
31	011x x xxx	Operand 1 address	
71	011x x xxx	Op 1 disp from XR1	
B1	011x x xxx	Op 1 disp from XR2	

DA M N

N-Code	To Be Loaded
000	Load byte count register. (The byte count value should equal the number of bytes to be transferred between the CPU and the tape subsystem.)
100	Load magnetic tape address register. (The address loaded should be the starting storage address for data to be written into or read from.)
110	*Model 10:* Invalid N-code; causes a processor check. *Models 8 and 12:* Load op-end indicator control register. *Model 15:* Load interrupt control register. (The storage byte specified by the effective address holds the control code; the other byte is not used.)

Code Bits 1234 5678	*Model 15*	*Models 8 and 12*
xxx0 0xxx	Disable all tape subsystem op-end interrupts	Disable all tape subsystem op-end indicators
xxx1 0xxx	Enable all tape subsystem op-end interrupts	Enable all tape subsystem op-end indicators
xxx0 1xxx	Reset addressed tape unit op-end interrupt	Reset address tape unit op-end indicator
xxx1 1xxx	Reset attachment op-end interrupt	Reset attachment op-end indicator

Any N-code not shown in invalid and causes:
- Program check if interrupt level 7 is enabled on Model 15
- Processor check if interrupt level 7 is not enabled on Model 15
- Processor check on Models 8, 10, and 12

Bits 01234	Tape Unit Specified
01100	0
01101	1
01110	2
01111	3

31, 71, or B1 specifies a load I/O operation. The first hex character in the op code specifies the type of operand addressing to be used for the instruction.

Operation

The processing unit moves the contents of the 2-byte field specified by the operand address to the register specified by the N-code. The operand is addressed by its low-order (higher numbered) position.

Program Notes

- The LIO instruction is accepted if the subsystem and addressed device are not busy. An LIO with an N-code of 110 is always accepted; all other LIO instructions are rejected if the subsystem or addressed device is busy.

- Any tape unit can be addressed to:
  - Enable or disable op-end interrupts or op-end indicators, or
  - Reset the attachment op-end interrupt or op-end indicator.

- A disable op-end interrupt instruction on Model 15 is effective immediately; that is, no op-end interrupt request will be issued for any operation in progress, but does not discontinue any interrupt in process when the instruction is issued. A disable op-end indicator instruction issued by a Model 8 or 12 resets all existing op-end indications on the tape subsystem.

- If interrupts are disabled, any interrupt that would have occurred as a result of an SIO is lost.

- An attachment interrupt or op-end indication, occurs for every SIO except those with N = 110 and 111 (CE diagnostic instructions). An attachment busy interrupt or op-end indication occurs for every SIO accepted by the tape attachment even if the instruction results in a no-op function. For this reason, no-op does not cause an additional interrupt or indication.

- A tape unit interrupt or op-end indication occurs at the completion of an SIO rewind operation or an SIO data security erase operation. If an SIO rewind is issued when the tape is at load point, only an attachment interrupt or operation finished indication occurs.

- If the tape is loaded and the manual reset-rewind-start sequence is performed, the tape unit goes busy until the rewind is complete, at which time the device goes ready and the attachment presents the appropriate tape unit interrupt request to the CPU. Because no SIO was issued, the attachment does not present an attachment interrupt request to the processing unit at this time.

3410/3411 TEST I/O AND BRANCH (TIO)

Op Code (hex)	Q-Byte (binary)	Operand Address	
Byte 1	Byte 2	Byte 3	Byte 4
C1	011x x xxx	Operand 1 address	
D1	011x x xxx	Op 1 disp from XR1	
E1	011x x xxx	Op 1 disp from XR2	

DA M N

N-Code	Condition
000	Not ready/check. (This condition occurs whenever the addressed tape unit becomes not ready. The condition is removed when the reason for the not-ready condition is corrected.)
001	*Model 8:* Op-end indicator on. *Model 10:* Invalid N-code; causes processor check. *Model 12:* Op-end indicator on. *Model 15:* Interrupt pending.
010	Busy. (This condition occurs for all addresses when the subsystem is executing a command. It also occurs for each addressed tape unit whenever that tape unit is executing a rewind of a data security erase operation, and whenever the addressed tape unit is executing a rewind/unload command.)

Any N-code not shown is invalid and causes:
- Program check if interrupt level 7 is enabled on Model 15
- Processor check if interrupt level 7 is not enabled on Model 15
- Processor check on Models 8, 10, and 12

Bits 01234	Tape Unit Specified
01100	0
01101	1
01110	2
01111	3

Hex C1, D1, or E1 specifies a TIO operation. The first hex character in the op code indicates the operand addressing scheme to be used for the instruction.

Operation

The CPU tests the drive specified by the DA and M-codes for the condition specified by the N-code. If the condition exists, the program branches to the address in the operand address portion of the instfuction. If the condition does not exist, the program proceeds with the next sequential instruction.

Program Note

An interrupt pending or op-end indicator on condition indicates that either the attachment itself or one of the tape units requires program action. To determine which unit caused the request, issue a sense instruction to test sense byte 0.

3410/3411 ADVANCE PROGRAM LEVEL (APL)

Op Code (hex)	Q-Byte (binary)	R-Byte (binary)
Byte 1	Byte 2	Byte 3
F1	xxxx x xxx	0000 0000

DA M N

R-byte is not used in an APL instruction.

N-Code	Condition Tested
000	Not ready/check. (This condition occurs, whenever the addressed tape unit becomes not ready. The condition is removed when the reason for the not-ready condition is corrected.)
001	*Model 8:* Op-end indicator on. *Model 10:* Invalid N-code; causes processor check. *Model 12:* Op-end indicator on. *Model 15:* Interrupt pending. (This condition indicates that either the attachment itself or one of the tape units requires program action. To determine which unit caused the request, issue a sense instruction to test sense byte 0.)
010	Busy. (This condition occurs for all addresses when the subsystem is executing a command. It also occurs for each addressed tape unit whenever that tape unit is executing a rewind or data erase operation, and whenever the addressed tape unit is executing a rewind/unload command.)

Any N-code not shown is invalid and causes:
- Program check if interrupt level 7 is enabled on Model 15
- Processor check if interrupt level 7 is not enabled on Model 15
- Processor check on Models 8, 10, and 12

Bits 01234	Tape Unit Specified
01100	0
01101	1
01110	2
01111	3

If the DA and M-codes = 00000, the APL instruction is treated as a no-op command and the processing unit immediately accesses the next sequential instruction. For this unconditional no-op, the N-code should be binary 000, but the program notes for this command in the processing unit section of the manual discuss what happens if the N-code is not 000.

Hex F1 specifies an advance program level operation. Hex F indicates that the instruction is a command-type instruction (no operand addressing).

Operation

This instruction tests for the conditions specified in the Q-byte.

- Condition present:
  - Systems with Dual Program Feature installed and enabled, activate the inactive program level.
  - Systems without Dual Program Feature installed or with Dual Program Feature installed but not enabled, loop on the advance program level instruction until the condition no longer exists.

- Condition not present: Systems with or without Dual Program Feature access the next sequential instruction in the active program level.

Program Note

For additional information concerning the advance program level instruction, see Chapter 2.

3410/3411 SENSE I/O (SNS)

Op Code (hex)	Q-Byte (binary)	Operand Address	
Byte 1	Byte 2	Byte 3	Byte 4
30	011x x xxx	Operand 1 address	
70	011x x xxx	Op 1 disp from XR1	
B0	011x x xxx	Op 1 disp from XR2	

DA M N

N-Code	Sensed Unit
000	Subsystem bytes 0[1] and 1[2]
001	Subsystem bytes 2[1] and 3[2]
010	Subsystem bytes 4[1] and 5[2]
011	Subsystem bytes 6[1] and 7[2]
100	Magnetic tape address register content
101	Attachment sense bytes
110	Subsystem hardware error sense
111	Reserved byte[2] and op-end sense[1]

Bits 01234	Tape Unit Specified
01100	0
01101	1
01110	2
01111	3

30, 70, or B0 specifies a sense I/O operation. The first hex character in the op code specifies the type of operand addressing for the instruction.

[1] Byte is stored at operand address minus 1.
[2] Byte is stored at operand address.

Operation

The processing unit transfers 2 bytes of data from the unit specified by the N-code to the main storage field specified by the operand address. The first byte enters the effective address (the operand address), the second byte enters the effective address minus 1. Status bits are described in Figure 6-8.

Program Notes

- Tape error conditions and general status information about the tape are conveyed to the processing unit as bits in sense bytes. Figure 6-7 shows program significant sense byte information; Figure 6-8 shows conditions that set bits in sense bytes 1, 2, 3, and 5.

- A valid sense instruction is always executed, either by the attachment or the subsystem. A complete sense operation must be performed every time the not-ready/unit check indication is on. A complete sense operation includes the 2 attachment sense bytes and all 8 subsystem sense bytes. To perform a complete sense operation, an SNS instruction must be issued to each of the four Q-byte configurations that define subsystem sense bytes. An SNS instruction does not reset any sense or status information (except no-op when sense byte 0 is sensed).

- An SNS instruction causes the subsystem to request an I/O cycle steal after the Q-byte. This provides time for the subsystem to assemble the requested sense information. No data is transferred during the I/O cycle, and the MTAR does not change.

- Before the instruction is performed, the MTAR must have proper parity and the address must be less than the system's main storage size. Improper parity, or an address equal to or larger than the main storage size, causes a processor check stop when the MTAR is used during this I/O cycle.

- Sense information is defined by byte 0, bit 7 of the attachment and subsystem sense bytes. Sensing a tape subsystem that is busy (performing a command) or incapable (hardware error) forces attachment sense bytes (CE sense bits) to the CPU in place of the subsystem sense bytes requested. This is indicated by bit 7 being off. The subsystem sense was performed properly if bit 7 is on.

Byte	Bit	Name	Indicates
0	0	Noise	(1) A block of data read in NRZI mode was less than 13 bytes long and a data check occurred, or (2) signals were detected during the read back check of an erase gap operation (PE or NRZI).
0	1	Wrong length record	The number of bytes in a block is different from the processing unit byte count.
0	2	Unit exception	A tape unit detected (1) an end-of-tape marker during a write operation, or (2) a tape mark during a read forward or space block operation.
0	3	Data check	An error that allows a retry after proper tape positioning occurred.
0	4	Diagnostic track error	
0	5	No-op	The attachment accepted a command that it could not execute.
0	6	Equipment check	(1) The addressed tape unit did not receive a command addressed to it, (2) an unknown tape position exists, or (3) a tape mark could not be written properly.
0	7	Sense valid	The subsystem sense was performed properly. (If the bit is off, subsystem sense bytes 0 and 1 are replaced by attachment sense bytes 0 and 1.)
1	0	Data converter check	The data block read during a seven-track read forward operation did not contain a multiple of 4 bytes. This condition also sets the data check bit.
1	1	Command reject	The program issued (1) a write, write tape mark, or erase gap command to a file-protected tape unit or (2) a data security erase command to a tape unit in read status. This condition sets the no-op check bit.
1	2	Backward at load point	(1) The tape entered the load pointer or (2) the program issued a command with tape at load point during a backward operation.
1	3	Start velocity check	Selected tape had not attained correct speed for recording data when data was ready to be written. This condition sets no-op sense byte.
1	4	Illegal command	The instruction contained an invalid control code (byte 3 of the instruction). See Figure 6-2 for valid control codes. This condition sets the no-op check bit.
1	5	Tape unit status changed	The attachment accepted the instruction but could not execute it because the addressed tape unit was not ready. This condition sets the no-op check bit.
1	6	Word count zero	The byte count was 0 at the start of a read, read backward, or write data operation. This condition sets the no-op check bit.
1	7	Not capable	During a read from load point, a PE identification was not detected on a tape mounted on a PE drive. This condition sets the no-op check bit.
2	0	Backward	These bits are not reserved.
2	1	Not file protected	
2	2	Tape indicate	The tape unit performed a write, erase gap, data security erase, or write tape mark operation when the tape was positioned at, or past, the end-of-tape reflective marker. This condition sets the unit exception bit.
2	3	Beginning of tape	
2	4	Write status	
2	5	Start key	

Figure 6-7 (Part 1 of 2). Tape Subsystem Sense Bytes

Byte	Bit	Name	Indicates
2	6	Tape unit check	An error occurred in the tape unit. This condition also sets the equipment check bit. Subsystem sense byte 6, bit 0, 1, 2, or 3 indicates the nature of the error.
2	7	Not busy	
3	0	Tape mark check	The tape unit tried unsuccessfully to write an acceptable tape mark. The subsystem automatically repositions the tape and retries the write tape mark operation up to 15 times. If the retries fail to write an acceptable tape mark, the subsystem sets the equipment check sense bit.
3	1	End velocity check	Tape was not moving at proper speed at the end of the read back check during a write operation. This condition sets the data check bit.
3	2	Tape unit position	During selection of a tape unit, the tape was still moving within the IBG when it was expected to be motionless. This condition sets the equipment check bit.
3	3	Reject tape unit	The selected tape unit failed to enter read status or write status, or became not ready during execution of a tape motion operation. This condition also sets the equipment check sense bit.
3	4	Write feed through	
3	5	No readback data	Data was not sensed at the read head during a write operation, so write checking could not be performed on the data. This condition also sets the equipment check sense bit.
3	6	Tachometer failure	Absence of tachometer pulses when the tape should be in motion. This condition also sets the equipment check sense bit.
3	7	Overrun	The processing unit cannot grant cycle steals fast enough to transfer data without loss of bytes. Data transfer stops and the equipment check bit is set when this condition occurs.
5	0	Attachment bus out check	Data from the processing unit during a write operation had even parity. This condition also sets the data check sense bit.
5	1	Multitrack/longitudinal redundancy check (LRC)	A tape unit in PE mode had envelope dropout in 2 or more tracks and/or a phase error after a read, write, or read backward operation. It also indicates that the LRC register is not 0 or has incorrect parity after a read, write, or read backward operation in NRZI mode. These conditions also set the data check bit.
5	2	Data timing check	Bits within a data byte are excessively misaligned during a read or read backward operation in PE mode or during a write operation in NRZI mode. These conditions also set the data check sense bit.
5	3	End data/cyclic redundancy check (CRC)	*In PE mode,* the tape unit did not detect the ending burst of bits following a data block during a read or read backward operation. *In NRZI mode* (1) the CRC byte read from the tape did not match the CRC pattern generated by the subsystem while the data block was read (2) during write operations the CRC byte parity had incorrect parity on the read-back check, or (3) during write operations read-back checking, the CRC pattern did not match the pattern that was written. These conditions also set the data check bit.
5	4	Envelope check or phase error	An envelope check or phase error occurred during a read, read backward, or write operation in PE mode. These conditions also set the data check bit if they occur during write operations. During read and read backward operations, data check is set only if an uncorrectable error occurred with an envelope check or phase error.
5	5	False end marker	The subsystem detected an incorrect end-of-tape marker during a read or write operation. This condition also sets the data check sense bit.
5	6	PE ID-burst check	*In PE mode,* (1) the PE identification burst was improperly written or (2) a start velocity error occurred during write operations. *In NRZI mode,* a start velocity error occurred during a write operation from load point. These conditions also set the data check sense bit.
5	7	Vertical redundancy check (VRC)	A parity error was detected (1) on data being read during read and read backward operations and the error was not corrected, or (2) on data being read during readback checking during write operations. These conditions also set the data check sense bit.

Figure 6-7 (Part 2 of 2). Tape Subsystem Sense Bytes

Sense Byte	Sense Bit	Condition[1]	Sense Byte 0 Bits: Unit Exception Bit 2	Data Check Bit 3	No-op Bit 5	Equipment Check Bit 6
1	0	Data converter check		X		
	1	Command reject			X	
	2	Backward at load point			X	
	3	Start velocity check			X	
	4	Illegal command			X	
	5	Tape unit status changed			X	
	6	Word count zero			X	
	7	Not capable			X	
2	2	Tape indicate	X			
	6	Tape unit check				X
3	0	Tape mark check				X
	1	End velocity check		X		
	2	Tape unit position				X
	3	Reject tape unit				X
	5	No readback data				X
	6	Tachometer failure				X
	7	Overrun		X		
5	0	Attachment bus out check		X		
	1	Multitrack/LRC		X		
	2	Data timing check		X		
	3	End data/CRC		X		
	4	Envelope check		X		
	5	False end marker		X		
	6	PE ID burst check		X		
	7	Vertical redundancy check		X		

[1]These conditions set sense byte 0 bits designated by an X.

Figure 6-8. Tape Subsystem Sense Information

Byte	Bit	Name	Indicates	Reset By
Op-end	0	Tape unit 0 op-end	Tape unit 0 completed its operation and requires program action.	1. Issuing a disable op-end interrupt LIO (only pending interrupts are reset; current interrupt being serviced is not reset), or 2. Issuing a reset tape unit op end interrupt LIO addressing the tape unit whose op end bit is to be reset
Op-end	1	Tape unit 1 op-end	Tape unit 1 completed its operation and requires program action.	
Op-end	2	Tape unit 2 op-end	Tape unit 2 completed its operation and requires program action.	
Op-end	3	Tape unit 3 op-end	Tape unit 3 completed its operation and requires program action.	
Op-end	4	Subsystem op-end	The subsystem control unit completed its operation and requires program action.	1. Issuing a disable op end interrupt LIO (only pending interrupts are reset; current interrupt being serviced is not reset), or 2. Issuing a reset subsystem op-end interrupt LIO addressing any of the tape units.
Op-end	5	Reserved		
Op-end	6			
Op-end	7			

Figure 6-9. Tape Op-End Status Byte for Models 8, 12, and 15

Chapter 7. Disk Storage Drives

IBM 5444/5448 Disk Storage Drives

IBM 5444 DISK STORAGE DRIVE

The IBM 5444 Disk Storage Drive provides 2,457,600 through 9,830,400 bytes of storage.

The 5444 is available in six models:

Model	Tracks/ Surface[1]	No. of Disks[2]	Total Capacity (in bytes)	Avg Access Time
1	104	2	2,457,600	153 ms
A1	104	2	2,457,600	86 ms
2	204	2	4,915,200	269 ms
A2	204	2	4,915,200	126 ms
3	204	1	2,457,600	269 ms
A3	204	1	2,457,600	126 ms

[1] IBM resident control program requires one track per surface for customer engineers. IBM disk systems programming support requires 3 tracks per surface for alternate data tracks. Systems using these programs are therefore limited to 100 or 200 tracks per surface, according to the model selected.

[2] Each model has one removable disk (Figure 7-1). Models 1, A1, 2, and A2 also have 1 permanent disk. Both surfaces of each disk are used.

All six models of the 5444 Disk Storage Drive perform at a data rate of 199 kilobytes per second (±5%) at a rotation speed of 1500 rpm.

The 5444 can be ordered with the following configurations of models:

- System/3 Model 8
  - One Model A1
  - One Model A2
  - Two Model A2s
  - One Model A2 and one Model A3
- System/3 Model 10

Standard Speed Access	*High Speed Access*
One Model 1	One Model A1
One Model 2	One Model A2
Two Model 2s	Two Model A2s
One Model 2 and one Model 3	One Model A2 and one Model A3

- System/3 Model 15A
  - One Model A2
  - Two Model A2s
  - One Model A2 and one Model A3

REMOVABLE DISK CARTRIDGES FOR 5444

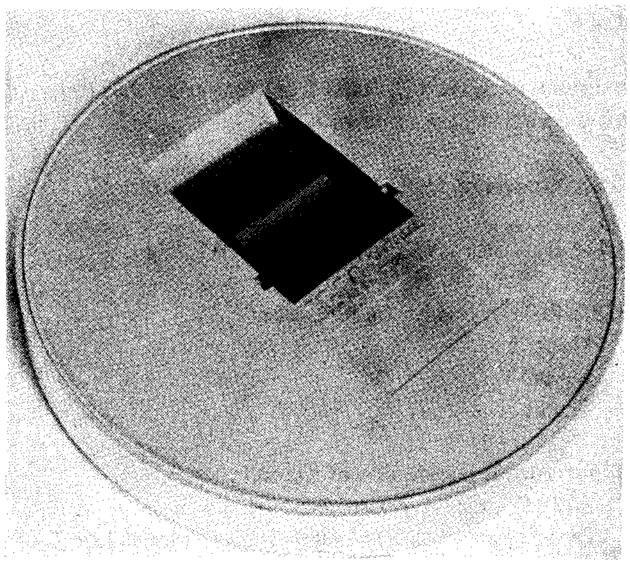


Figure 7-1. IBM 5440 Disk Cartridge

Each 5444 uses a removable IBM 5440 Disk Cartridge (Figure 7-1). Using removable cartridges provides virtually unlimited offline disk storage and allows data interchange between all 5444 models on all IBM System/3 models. However, data recorded on the second 100 cylinders of a 5440 by an IBM 5444 Model 2, A2, 3, or A3 cannot be read by a 5444 Model 1 or A1. Also, a 5440 disk initialized on a 5444 Model 1 or A1 is not initialized on the second 100 cylinders.

Care and handling procedures for the 5440 are described in *IBM 5440 Disk Cartridge Handling Procedure Manual,* GA26-1598.

IBM 5448 DISK STORAGE DRIVE

The IBM 5448 Disk Storage Drive provides an additional 9,830,400 bytes of storage for the System/3 Model 8 and Model 10. It is available in one model (A1).

The 5448 is housed in its own enclosure and contains two disk drives identically designed. Each drive has two fixed disks attached to a common spindle and is capable of storing 4,915,200 bytes with an average access time of 126 ms. Each disk has 408 tracks (204 on each surface). The IBM resident control program requires one track per surface reserved for customer engineers. IBM disk systems programming support requires three tracks per surface for alternate data tracks. Systems using these programs are, therefore, limited to 200 tracks per surface.

SYSTEM CONFIGURATION

Model 8

The minimum System/3 Model 8 configuration that the 5448 is designed to operate with is:

- 5408 Processing Unit, Model A14 (16K bytes)
- 5444 Disk Storage Drive, Model A2 (R1, F1)
- 5203 Printer, one of the following:
  Model 1
  Model 2
  Model 3
- Input/output device, one of the following:
  5471 Printer-Keyboard, Model 1
  Directly attached 3741 Data Station, Model 1 or 2
  Directly attached 3741 Programmable Work Station, Model 3 or 4

Note: The 5448 cannot be attached to a Model 8 with the Serial Input/Output Channel (SIOC).

Model 10 Disk System

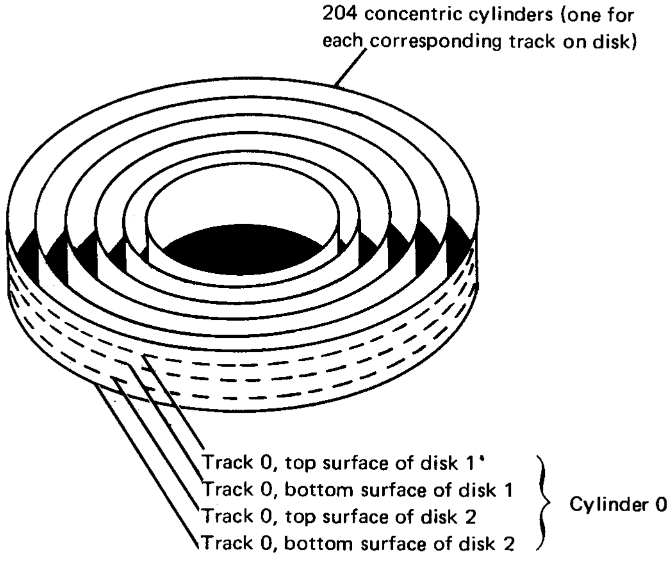
The minimum System/3 Model 10 Disk System configuration that the 5448 is designed to operate with is:

- 5410 Processing Unit, Model A13 (12K bytes)
- 5444 Disk Storage Drive, Model 2 or A2 (R1, F1)
- Printer, one of the following:
  5203 Printer, Model 1, 2, or 3
  1403 Printer, Model 2 or N1
- Input/output device, one of the following:
  5424 MFCU, Model A1 or A2
  1442 Card Read Punch, Model 6 or 7

Note: The 5448 cannot be attached to a Model 10 Disk System with the 5445 Disk Storage.

5444/5448 DISK ORGANIZATION

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 7-2. On drives with two disks, the corresponding cylinders on both disks have the same cylinder number.



Note: The same cylinder address is used, for all corresponding tracks on the disks. For example, track 15 on both the upper and lower surfaces of disks 1 and 2 are all considered to be bands of data on one cylinder, so all four bands have the same cylinder address. On the 5444/5448, the same track on both the upper and lower surfaces of a single disk are considered to be a cylinder.

Figure 7-2. 5444 Cylinder Concept

5444/5448 Track Format

Each track is divided into 24 sectors (Figure 7-3). Each sector has an individual address. A sector contains:

- *Index Marker* – A mark that is fixed for each disk and provides orientation information to the controlling circuits. It is the starting point for every track.
- *AM* – Address marker is a specially written group of bits used to indicate the start of a new sector.
- *ID* – The sector identifier. This group of 6 bytes contains 3 bytes for unique identification of that sector for that disk, and 3 bytes of check characters.
- *Data* – The data area of the sector contains 256 bytes of data and 3 bytes of check characters.
- *Gaps* – Gaps are specially written areas on the disk used to separate and define the other elements of the sector.

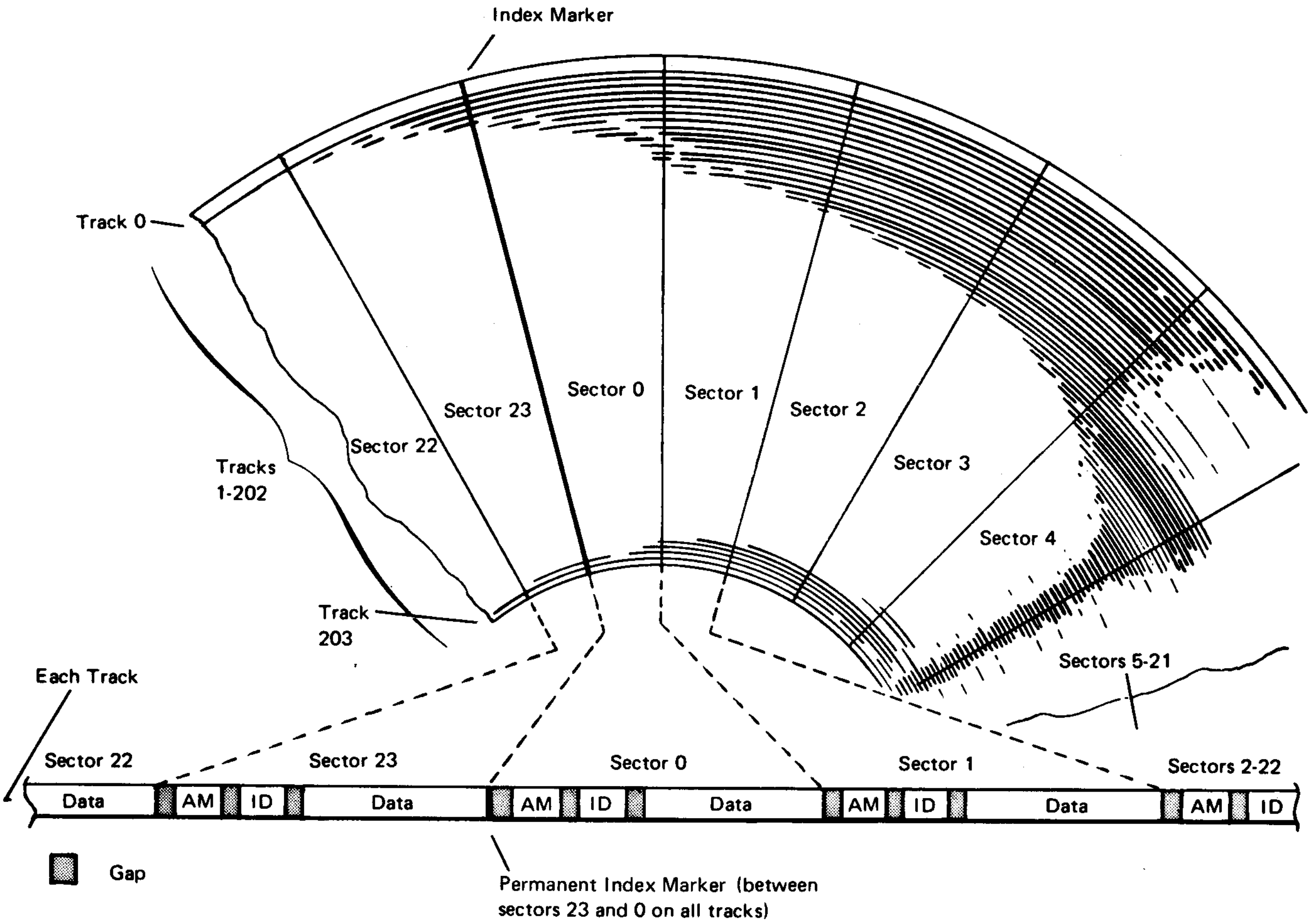


Figure 7-3. Sector Layout

5444/5448 Sector Identifier Format and Addressing

The identifier area of a sector (ID) contains a flag byte, 2 bytes of address information, and 3 bytes of check information as shown below:

Flag	Address		Check Characters		
F	C	S	CC	CC	BCA

F Flag byte. This byte contains flagging information in bits 6 and 7. All other bits in this byte should be 0.

C Cylinder byte. This byte contains the binary number that corresponds to the physical location of the track on the disk.

S Sector byte. The 6 leftmost bits in this byte represent the binary number of the sector. Sectors on the upper 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. The attachment generates these 2 bytes and uses them for checking purposes.

BCA Bit count appendage. Another checking byte the attachment generates.

The address of any individual sector is contained in the second and third bytes of the identifier. Sectors occupying the same physical location on the lower disk and on the upper (removable on the 5444) disks *have identical binary numbers in the cylinder and sector bytes.* Use of a sector requires that the drive (1 or 2) and the disk (lower or upper [removable on the 5444]) containing the desired sector must be specified.

Cylinders are numbered 0 through 203, counting from the outer cylinder. IBM customer engineers use cylinder 203 for diagnostic functions, so this cylinder should not be used for permanent storage. Tracks in cylinders 1, 2, and 3 are used by IBM program products as alternate tracks if tracks in cylinders 1 through 202 become defective; therefore, if IBM program products are used, cylinders 1, 2, and 3 are reserved. Tracks in cylinders 0 and 4 through 202 can be used as standard data tracks.

Sectors within a track are identified by their physical position on the track with relation to the index marker and by the surface of the disk on which they reside. The sectors on the upper surface of the disk are numbered 0 through 23 starting from the index marker, and the sectors on the lower surface are numbered 32 through 55. A specific sector address, then, consists of a drive number (upper or lower disk), a cylinder number, and the sector number. However, only the cylinder number and sector number are recorded on the disk.

5444 (ONLY) DISK OPERATING RESTRICTIONS

The disk drive drawers cannot be opened unless system power is on and the disk start/stop switch on the system control panel is in the stop position. The OPEN light on the system control panel lights when it is safe to open the drawers. We recommend that the drawers be kept closed unless a disk cartridge is being inserted or removed. A cartridge should always be stored on the drive to prevent dust from entering the drive.

The 5440 disk cartridge must be stored in the operating environment for at least 2 hours before the cartridge is used for processing.

5444/5448 DISK OPERATIONS

For each disk operation, the address of the disk control field must be stored in the disk control address register and the address of the first byte of the disk data field must be stored in the disk read/write address register.

The disk control field is 4 bytes long; these bytes are designated F-byte, C-byte, S-byte, and N-byte. The bytes are used as follows:

Byte Use

F This is the first byte in the field and the byte addressed by the disk control address register. In seek operations, this byte is not used. In other disk operations, it contains flag bits in bits 6 and 7.

C This second byte of the field contains a binary number that designates a cylinder number. This byte is not used on a seek operation.

S The function of this byte (the third byte in the field) depends on the operation:

Seek Operation: Bit 0 selects the head to be used (0 = head 0 for upper surface; 1 = head 1 for lower surface). Bits 1 through 6 are not used. Bit 7 selects direction of seek (0 = toward decreasing cylinder numbers; 1 = toward increasing cylinder numbers).

All Other Operations: Bits 0 through 5 hold the binary representation of the sector ID number. Bits 6 and 7 are not used; bit 7 must be 0.

N This last byte in the field specifies either the number of cylinders to move the access mechanism for a seek operation or the number of sectors to use for any other operation. For operations other than seek, this binary number must be 1 less than the actual number of sectors desired. For example, if 1 sector is to be used, the N-byte must contain a 0; if 10 sectors are to be used (a multiple-sector operation), the N-byte must contain a 9.

5444/5448 Seek Operation

The access mechanism of the selected drive is moved a specified number of cylinders and the upper or lower head for the specified disk is set for future read, write, verify, or scan operations. The number of cylinders to cross and the head to set are specified by the disk control field as described before.

The N-byte specifies the number of cylinders the access mechanism travels during the seek.

Bit 7 of the S-byte specifies the direction of movement. Forward (bit 7 = 1) is from cylinder 0 to 202. The head is specified by bit 0 of the S-byte.

The recalibration function is executed by specifying a seek in the reverse direction and the number of cylinders to move (greater than or equal to 224). The recalibrate function causes the access mechanism to return to cylinder 0 and selects read/write head 0, regardless of the condition of bit 0 of the S-byte.

Note: On high performance disk drives, recalibration should be used only for error recovery, because recalibration forces a low speed seek in a reverse direction.

The C-byte, 0 bit in the sense byte is set when the mechanism reaches cylinder 0, and can be interrogated by the program with a sense I/O instruction after the seek is complete.

Seek operation is begun by issuing an SIO instruction. A second SIO instruction can be issued to the same disk drive if a read, write, or scan operation is specified. The second instruction is accepted provisionally and executed if no errors occur in the operation of the seek instruction. A subsequent SIO instruction to either disk causes the CPU to loop on that instruction until the read, write, or scan operation ends. However, seek commands to both drives can be executed concurrently if there is no intervening SIO read, write, or scan instruction.

No data in storage is changed by this operation. Test I/O for busy or advance program level on busy does not detect busy unless a read, write, or scan instruction has been provisionally accepted. The sense bit for seek busy is on, however, for interrogation by the sense I/O instruction.

A seek to the cylinder at which the access mechanism is located is completed immediately because no access mechanism motion is required. However, the head is selected according to bit 0 of the S-byte.

5444/5448 Access Time

Access time is the interval from the receipt of a seek command until read/write head movement stops.

Access Time for the 5444 Models 1, 2, and 3 Only: Figure 7-4 shows the approximate time required to seek across any number of cylinders from 1 to 200. Access time can also be determined from the following formula:

Seek time for 1 cylinder = 39 milliseconds
Seek time for 2 or more cylinders = 47 + 3.42 (N-2) milliseconds

where N = the number of cylinders to be crossed.

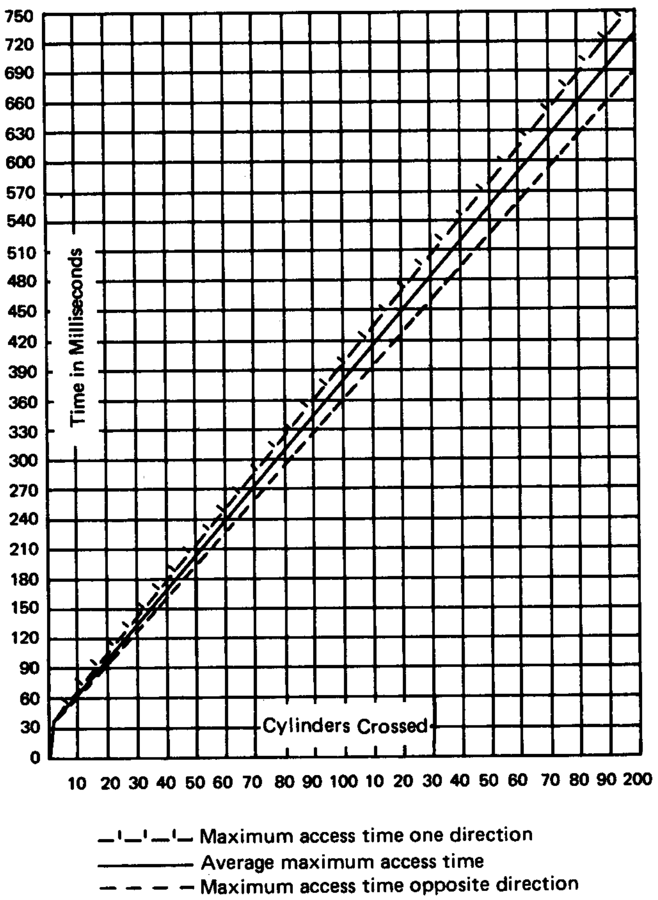


Figure 7-4. Access Timing (5444 Models 1, 2, and 3 only)

Access Time for 5444 Models A1, A2, A3, and 5448: For the high performance disk storage drive models, access times are not necessarily the same for both forward and reverse seek operations. The more important access times (for both forward and reverse directions) for these disk drives are:

- The access time for a 1-cylinder movement is 28 milliseconds for all three models.
- The normal average access time for a 5444 Model A1 is 86 milliseconds; for 5444 Models A2, A3, and 5448, the normal average access time is 126 milliseconds. This is the average access time across 67 cylinder addresses with the exception of when a forward seek terminates in cylinder address 170 through 203.
- The maximum access time (the time taken for the access mechanism to cross the maximum number of cylinders available on each model) is 163 milliseconds for 99 cylinders on Model A1; for 5444 Models A2, A3, and 5448, the access time is 255 milliseconds for 199 cylinders.

To Determine Approximate Maximum Access Forward Time: Access times for access forward operations are not dependent solely on the number of cylinders traveled, but also depend on where the access forward operations terminate. For this reason, no simple graph can be drawn showing access times for all possible access forward operations.

Figure 7-5 shows maximum access time curves for access forward operations starting from several different cylinder addresses. Each curve is labeled with its appropriate starting cylinder address. Intermediate values may be determined by interpolation.

To determine the access time for any forward operation follow the curve corresponding to the starting cylinder address until the curve coincides with the cylinder address that is being accessed (horizontal axis). The corresponding access time is then read from the vertical axis in milliseconds. For example, to determine the access time for an access operation from cylinder address 040 to cylinder address 120, follow the curve corresponding to cylinder address 040 until the curve is aligned with cylinder address 120 on the horizontal axis. The required access time indicated on the vertical axis is 140 milliseconds.

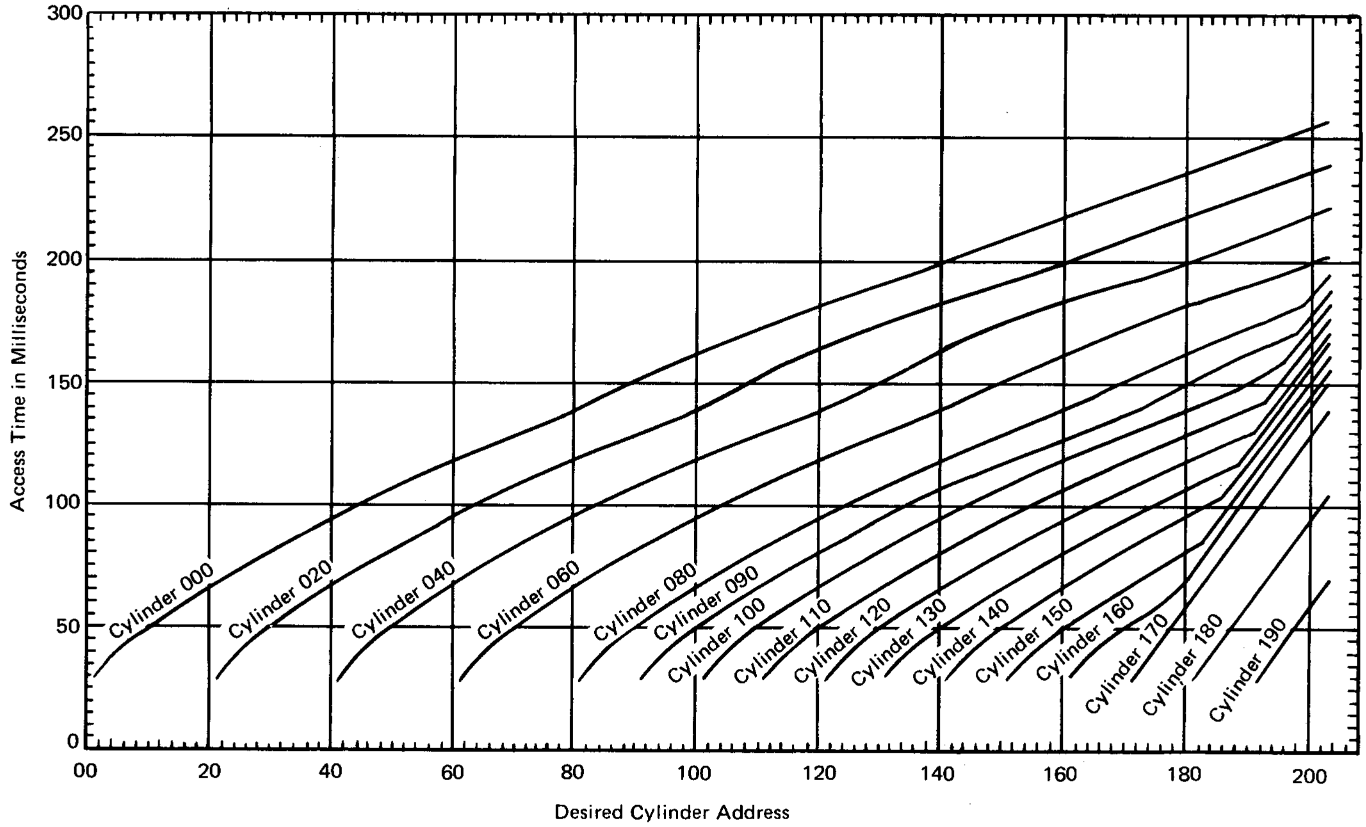


Figure 7-5. Maximum Access Time for 5444 Models A1, A2, A3, and 5448 (Forward Direction)

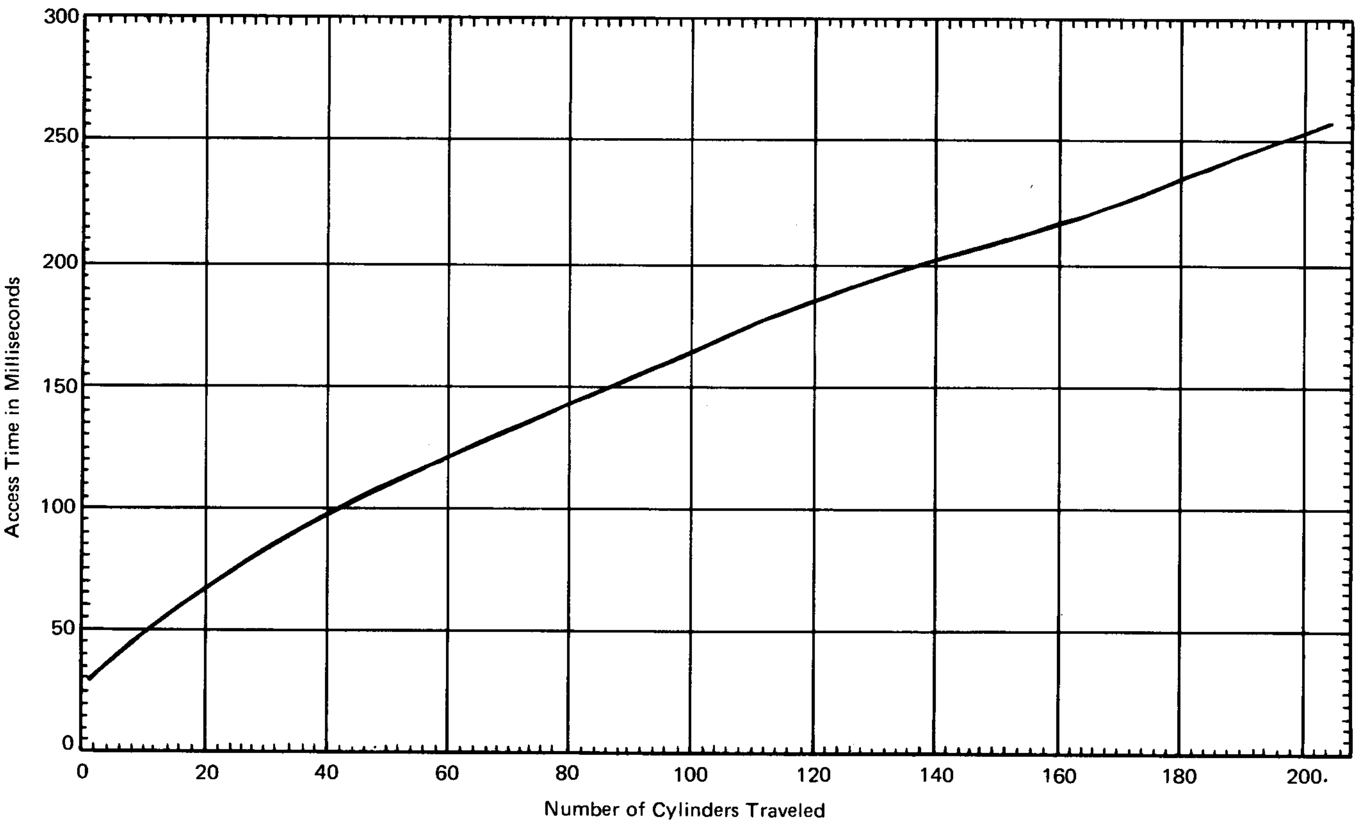


Figure 7-6. Maximum Access Time for 5444 Models A1, A2, A3, and 5448 (Forward or Reverse Direction)

To Calculate Maximum 5444/5448 Access Forward Time: Approximate maximum access times for access forward operations are shown in Figure 7-6. However, some access forward operations finishing above cylinder address 170 have access times greater than that indicated in Figure 7-6. (Reverse operations are not affected.)

Figure 7-7 shows all possible access operations. The chart is divided by a diagonal line into two regions covering access forward operations and access reverse operations. The access forward region is further subdivided into three areas.

To determine the maximum access time for any access forward operation, find the point where the from cylinder address (horizontal axis) and the to cylinder address (vertical axis) intersect. The area in which this point of intersection occurs defines how the access time is calculated, as follows:

1. Unshaded area—access time determined directly from Figure 7-6.

2. Shaded area—access time determined by Figure 7-6 plus an additional time as indicated by the chart (Figure 7-8).

3. Cross-hatched area—access time shown in Figure 7-9.

For example, to determine the maximum access time for an access operation from cylinder address 150 to cylinder address 200:

1. From Figure 7-7, locate the point of intersection of the present cylinder address (cylinder address 150) and the new cylinder address (cylinder address 200). The point of intersection is in the shaded area.

2. From Figure 7-6, determine that maximum access time for a 50-cylinder address difference is 107 milliseconds.

3. From Figure 7-8, determine that the additional time to be added is 39 milliseconds.

Therefore, the total maximum access time for this access operation is 146 milliseconds (107 + 39 = 146 milliseconds).

To Calculate Access Reverse Time: Figure 7-6 shows the maximum access time for the number of cylinders that the access mechanism crosses during an access reverse operation.

Note: Ready may be dropped if an access reverse operation specifies more tracks than the actual number of tracks from the present track to the home position. If ready is dropped and no permanent hardware fault exists, stop the disk drive and then restart the disk drive to establish a file ready condition.

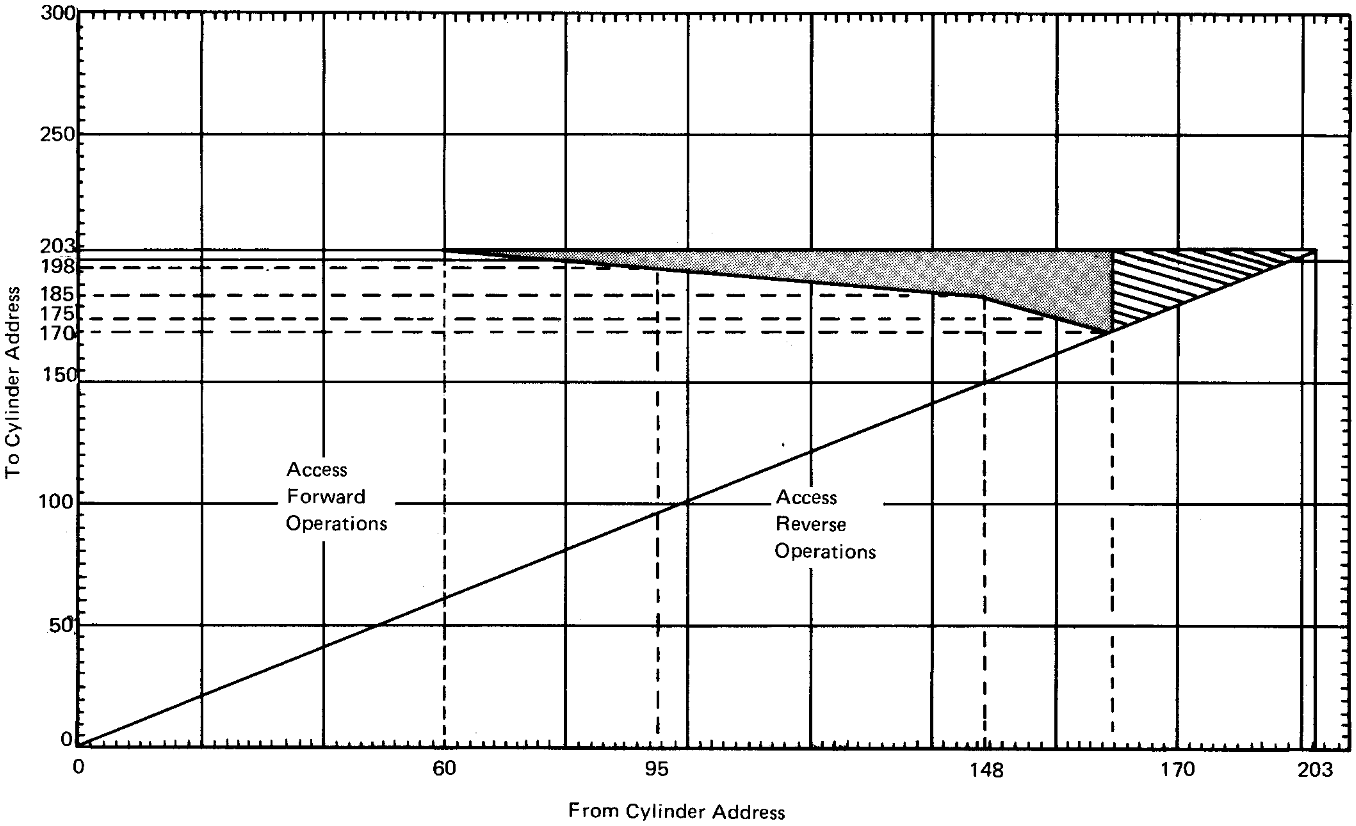


Figure 7-7. Access Time Chart for 5444 Models A1, A2, A3, and 5448

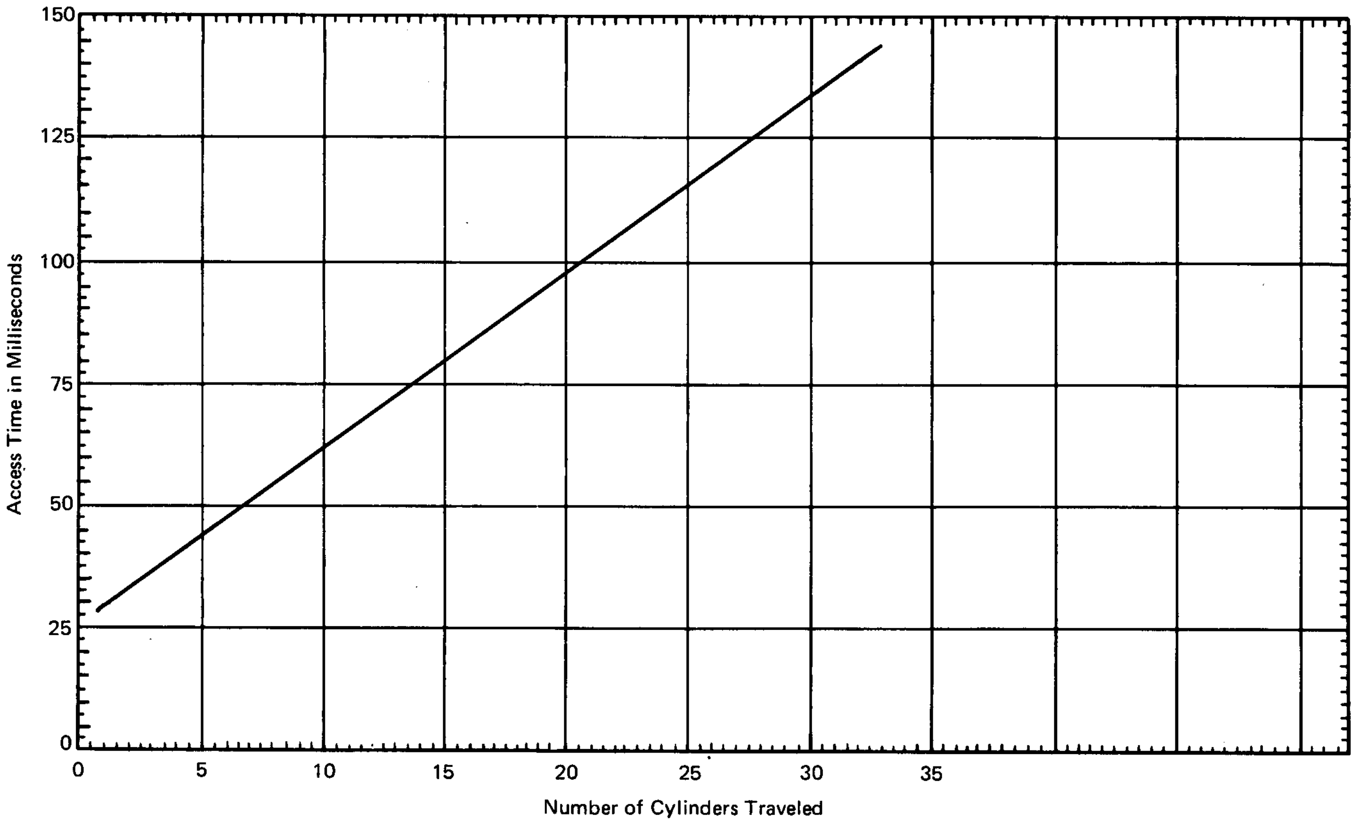
From Cylinder Address

To Cylinder Address	60	68	76	84	92	96	100	104	108	112	116	120	124	128	132	136	140	144	148	150	152	154	156	158	160	162	164	166	168	169
203	3	3	6	9	12	15	15	18	21	24	27	27	30	33	36	39	39	42	45	45	48	48	51	51	51	54	54	57	57	60
202	0	3	3	6	9	12	15	15	18	21	24	27	27	30	33	36	39	39	42	45	45	45	48	48	51	51	51	54	57	57
201		0	3	3	6	9	12	15	15	18	21	24	27	27	30	33	36	39	39	42	42	45	45	45	48	48	51	51	54	54
200			0	3	3	6	9	12	15	15	18	21	24	27	27	30	33	36	39	39	39	42	42	45	45	45	48	51	51	54
199				0	3	3	6	9	12	15	15	18	21	24	27	27	30	33	36	36	39	39	39	42	42	45	45	48	51	51
198					0	3	3	6	9	12	15	15	18	21	24	27	27	30	33	33	36	36	39	39	39	42	45	45	48	48
197						0	3	3	6	9	12	15	15	18	21	24	27	27	30	33	33	33	36	36	39	39	42	45	45	48
196							0	3	3	6	9	12	15	15	18	21	24	27	27	30	30	33	33	33	36	39	39	42	45	45
195								0	3	3	6	9	12	15	15	18	21	24	27	27	27	30	30	33	33	36	39	39	42	42
194									0	3	3	6	9	12	15	15	18	21	24	24	27	27	27	30	33	33	36	39	39	42
193										0	3	3	6	9	12	15	15	18	21	21	24	24	27	27	30	33	33	36	39	39
192											0	3	3	6	9	12	15	15	18	21	21	21	24	27	27	30	33	33	36	36
191												0	3	3	6	9	12	15	15	18	18	21	21	24	27	27	30	33	33	36
190													0	3	3	6	9	12	15	15	15	18	21	21	24	27	27	30	33	33
189														0	3	3	6	9	12	12	15	15	18	21	21	24	27	27	30	30
188															0	3	3	6	9	9	12	15	15	18	21	21	24	27	27	30
187																0	3	3	6	9	9	12	15	15	18	21	21	24	27	27
186																	0	3	3	6	9	9	12	15	15	18	21	21	24	24
185																		0	3	3	6	9	9	12	15	15	18	21	21	24
184																			0	3	3	6	9	9	12	15	15	18	21	21
183																				0	3	3	6	9	9	12	15	15	18	18
182																					0	3	3	6	9	9	12	15	15	18
181																						0	3	3	6	9	9	12	15	15
180																							0	3	3	6	9	9	12	12
179																								0	3	3	6	9	9	12
178																									0	3	3	6	9	9
177																										0	3	3	6	6
176																											0	3	3	6
175																												0	3	3
174																													0	0

Note: To reduce the size of the chart, the from cylinder address axis is not continuous. If the required cylinder address is not listed, use the next higher cylinder address. In some cases, this will mean that the additional access time obtained from the chart is a maximum of 3 ms greater than the true additional access time.

The chart specifies the number of milliseconds that must be added to the access time given by the general curve in Figure 7-7, for the range of cylinder addresses indicated.

Figure 7-8. Additional Access Time Chart (5444 Models A1, A2, A3, and 5448)



For a one-cylinder access:

Maximum access time = 28 milliseconds

For an access of more than one cylinder address above cylinder 170:

Maximum access time in milliseconds = 32 + 3.42 (N-2)

$2 \leq N \leq 33$

N = Number of tracks to be crossed

Figure 7-9. Maximum Access Times for Accesses above 5444/5448 Cylinder 170 (Forward Direction)

5444/5448 Read Data Operation

This instruction transfers data from the selected disk to main storage. Data is transferred in multiples of 256 bytes (the contents of an individual disk sector).

If reading is started at sector 0, all 48 sectors from corresponding upper and lower tracks on the same disk can be read as the result of a single read operation. Only consecutive sectors are read when multiple sector reading is indicated.

Reading begins with the sector specified by the S-byte of the disk control field in main storage. (Bit 0 of the S-byte for this instruction does not select the head, but is used for checking only; head selection can be accomplished only by a seek operation.) The data is transferred to processing unit storage, starting at the processing unit storage address specified by the disk data address register. Succeeding bytes are stored in progressively higher locations, because the 5444/5448 automatically updates the disk data address register so that it points to the storage address where the next byte of data is to be stored.

When the N-byte of the disk control field specifies that more than 1 sector is to be read, the 5444/5448 automatically updates the S-byte of the disk control field each time a sector is read so that it contains the address of the next higher sector on that cylinder and disk. After the 5444/5448 has read sector 23 and stored its data, the 5444/5448 automatically switches heads to read sector 32 from the associated track on the lower surface of the disk (the other track on the same cylinder on that disk). The read operation then continues. (Sector addresses cannot overflow from disk to disk because each disk contains identical addresses for common cylinders; that is, for cylinders with the same track number.)

During read operations, the 5444/5448 compares the disk control field with the sector identifier fields on the disk track to find the first sector to be read. The comparison is repeated for each additional sector to be read. If the disk control field and the sector identifier field fail to match, the operation terminates after the data portion of that sector is transferred to main storage even if other sectors remain to be read.

Two other abnormal conditions cause termination of the reading operation. Reading is terminated at the end of any sector in which an error is detected or if the sector read is the last sector (sector 55) in the cylinder.

During a read operation, the attachment generates 2 cyclic check (CC) bytes and a 1-byte bit count appendage from the data that has been read, and compares these to the CC bytes and bit count appendage read back with the data, providing a data check for read errors. During multiple sector reads, the operation is terminated at the end of any sector in which an error is detected except that an equipment check causes immediate termination. The data portion of the error sector is stored in storage and the 5444/5448 disk data address register is updated.

The read operation ends when the N-byte of the disk control field reaches hex FF and the data from that sector has been transferred. The number in the N-byte is decremented by 1 at the beginning of each sector transferred.

At the end of the operation, the 4 bytes of the disk control field contain information about the progress of the operation. The number of sectors processed is equal to the original value of the N-byte minus the value of N at the end of the operation, unless all sectors requested were processed. If all sectors have been processed, the value of N at the end of the operation is hex FF. The S-byte of the disk control field at the end of the operation contains the identifier of the last sector processed unless there is a missing address marker on the disk or no sector could be found with an identifier that matched that in the disk control field. If no sector was processed, the S-byte in the disk control field is the S-byte of the first sector desired. If an address marker is missing and a sector was processed in a multisector operation, the S-byte in the disk control field is that of the sector that lacks an address marker.

The disk control unit is busy to all other operations except sense I/O during a read data operation.

5444/5448 Read Identifier Operation

This operation transfers the sector identifier (F-, C-, and S-bytes) from the selected disk to storage. The operation starts with the first identifier to come under the head after the instruction is executed. It transfers the first sector identifier it finds to the address designated by the disk control address register. If an error is found in this identifier, the next sector identifier is read and transferred to storage starting at the original address contained in the disk control address register. The operation is terminated by the transfer of the first sector identifier found without an error, or by no record found, or by equipment check.

The disk control unit is busy to any new operation except sense I/O while the read identifier operation is being performed.

The information contained in the disk control field at the beginning of this operation is not used but is destroyed by the information read in from the disk. At the termination of this operation, the first 3 bytes (F-, C-, and S-) of the disk control field contain the last sector identifier read from the disk. The last (N) byte of the disk control field is not changed. This operation does not switch reading between the upper and lower surfaces of the disk.

At the end of the operation, the disk control address register contains the original address unless there is an equipment check. With an equipment check, the contents of the register may or may not contain the original address.

5444/5448 Read Data Diagnostic Operation

This operation is similar to a read data operation. Reading always begins at the index marker. Up to 48 sectors *can* be read (the entire contents of the cylinder), but no more than 24 sectors *should* be read. Exceeding the 24-sector limit increases the chances of reading the wrong data field into storage. The data portion of the record is read and placed in storage beginning at the address specified in the disk data address register. One is subtracted from the N-byte and added to the S-byte of the disk control field for each sector read. The data address in the disk data address register is returned to its original value at the beginning of each sector so that successive data fields overlay each other in storage. The operation ends at the end of the sector in which the N-byte is reduced to hex FF, the end of the cylinder is detected, or equipment check is detected. (No other conditions terminate the operation.) When the operation is terminated, data from the last sector read is in the disk data field in main storage.

This operation functions with reduced address marker requirements so that data that cannot be read by a read data operation because of a missing address marker may possibly be recovered.

The original sector identifier in storage (F-, C-, and S-bytes of the disk control field) should be the identifier of the first record on the track, so that the identifier area in storage at the end of the operation contains the identifier of the last record read unless there is no record found without a data check or a track condition check. A no-record-found without a data check or a track-condition check indicates that an address marker is missing earlier on the track.

Errors that do not terminate the read operation are reset at the end of the sector in which they occur unless they occur in the last sector read.

The number of sectors read can be determined by subtracting the N-byte of the disk control field from the original value of the N-byte unless all sectors were read. If all sectors were read, the N-byte is set to hex FF.

The control unit is busy to any new operation except sense I/O while performing a read data diagnostic operation.

5444 (Only) Read IPL Operation

This operation is initiated by pressing LOAD on the system control panel. In order for the load key to cause initial program loading from disk (drive 1 only), the IPL selector switch on the system control panel must be set either to FIXED DISK or REMOVABLE DISK. The read IPL operation causes the 256 bytes of data contained in the first record after the index mark on track 0 of the selected disk to be transferred to storage starting with storage address 0000. Control is then passed to the processing unit to begin executing the instructions starting at address 0000.

No compare is made on the identifier of the first record. The first record found after the index mark is read and any error conditions are made available for program testing. If no record is found or the wrong record is read, the program will not start correctly. An unsuccessful IPL operation requires an operator retry.

5444/5448 Verify Operation

The verify operation is performed for write checking. It should be performed after every write operation to ensure data integrity. (If the write was a multiple-sector operation that crossed a track boundary, the head select must be reset to 0 by a seek operation before issuing the read verify instruction.)

This operation is performed in the same way as the read data operation except that no data is transferred to main storage, and the disk read/write address register is not updated. No cycle steals are required except for updating the sector and N-bytes in the disk control field.

The function of write checking is done by generating the cyclic check and bit count appendage characters from the data read from the disk and comparing them to the cyclic check and bit count appendage characters read from the disk.

At the end of the operation the disk control field contains information about the progress of the operation. The sector byte of the disk control field indicates the last sector verified. The number of sectors verified can be determined by subtracting the contents of the N-byte of the disk control field from the original value of the N-byte, unless all sectors were read. If all sectors were read, the N-byte contains hex FF.

5444/5448 Write Data Operation

This operation transfers data from storage to the selected track on the disk. Data is transferred in multiples of 256 bytes. The entire data contents of a cylinder can be written (48 sectors) if writing starts with head 0, sector 0. Only consecutive sectors can be written by multiple-sector write operations.

Writing begins with the sector specified by the identifier portion (F-, C-, and S-bytes) of the disk control field located in storage and addressed by the disk control address register. The identifier from the disk control field is compared with the sector identifiers read from the selected disk track. The head selection is the result of the last seek unless a multiple sector operation that caused a track boundary crossing was performed subsequently. The 5444/5448 automatically switches from head 0 to head 1 when it crosses the track boundary on the upper surface of the disk.

Comparing begins with the first sector identifier to come under the head. An equal condition between the disk control field identifier and the sector identifier enables the writing of the 256 bytes of data. The data is fetched from storage using the disk data address register for addressing.

When multiple sectors are indicated, 1 is added to the S-byte and 1 is subtracted from the N-byte of the disk control field for each sector written (except for switching heads, when 0 is added to the S-byte).

This updated disk control field identifier is then compared with the next identifier read from the disk. An equal comparison of all succeeding addresses must occur before their corresponding data fields are written on the disk. The data field of a sector is not written if an error is found before the writing of data begins.

The write data operation is terminated at the end of the sector in which the byte count (N-byte) was reduced to hex FF, the end of the cylinder is reached, or a check condition occurs. An equipment check terminates the operation immediately. The presence of an error can be determined by a test I/O and branch instruction.

The disk control unit is busy to any instruction except sense I/O while it is performing a write data operation.

During writing, the control unit generates two cyclic check characters and one bit count appendage character for each data field. The three characters are recorded at the end of the data field. To check for write errors, the program should initiate a verify operation.

At the end of the operation the disk control field contains information about the progress of the operation. The identifier portion of the disk control field indicates the last sector written or where writing was attempted. The number of records processed can be determined by subtracting the contents of the N-byte from the original value of the N-byte unless all sectors were written. If all sectors were written, the N-byte contains hex FF. If a track boundary is crossed in a multiple-sector write operation, head 1 remains selected.

5444/5448 Write Identifier Operation

This operation writes 24 sector formats (address marker, sector identifier, gaps, and data) on the selected track beginning at the index marker. There is no identifier field compare on a write identifier instruction before writing.

The identifier portion of the disk control field is written as the sector identifier of the first sector after the index marker. The N-byte of the disk control field is forced to a value of decimal 23 by this operation so that exactly 24 sectors are written on the track. As each identifier is written on the disk, 1 is added to the S-byte and 1 is subtracted from the N-byte of the disk control field.

The data field of each sector is filled with the characters stored at the address contained in the disk data address register. The register is *not* updated during the operation, so the same character is propagated in all data byte positions of all the sectors. During writing of each identifier and data field, the control unit generates 2 cyclic check bytes and 1 bit count appendage byte and automatically writes them as the last 3 bytes of both the identifier and the data fields. The check data for the identifier applies only to the identifier, and the check data for the data applies only to the data.

At the end of the operation, the disk control field contains information about the progress of the operation. The identifier portion of the disk control field indicates the last sector written or where writing was attempted. The number of records processed can be determined by subtracting the contents of the N-byte of the disk control field from 23, the original value of the N-byte, unless all records were processed. If all records were processed, the N-byte contains hex FF.

The disk control unit is busy to all new operations except sense I/O during a write identifier operation.

A verify operation must check for write errors following each write identifier operation to meet disk performance specifications.

5444/5448 Scan Operation

The scan operation searches the data fields on the disk to find one that meets certain specified conditions when compared with a 256-byte data field in storage. Up to 1 cylinder of data (48 sectors) can be scanned in one operation. The scan operation can specify one of the following conditions to satisfy the scan:

Equal
Low or equal
High or equal

The data in the sectors on the disk is compared with the 256 characters in the disk data field in storage. The disk data field is addressed by the 5444/5448 disk address register. The comparison of individual characters within the sector can be masked off by placing a mask character consisting of all bits (hex FF) in each noncompare byte in the disk data field in storage. If only 10 bytes are to be compared, the field must contain 246 mask characters in the byte positions of the characters that are not to be compared.

Scanning data begins with the sector specified by the identifier portion of the disk control field. Bit 0 of the S-byte for this instruction does not select the head but is used for comparison only. (Head selection can only be accomplished by a seek instruction.) Comparing sector addresses begins with the first sector identifier to come under the head. After the beginning sector is scanned, the S-byte is updated to the identifier of the next sector (by adding 1 to the S-byte value) and the N-byte is decreased by 1 for each sector scanned. The S-byte updating and head switching from 0 to 1 are automatic when a track boundary is crossed in a multiple sector operation.

The operation terminates under the following conditions:

1. When the data on the disk satisfies one of the following conditions specified by the start I/O instruction:
   a. Equal to the storage data field
   b. Equal to or lower than the storage data field
   c. Equal to or higher than the storage data field

2. At the end of the sector in which the sector count in the N-byte of the disk control field goes to FF

3. When the end of the cylinder is reached

4. At the end of any sector in which an error occurs after the first identifier specified by the disk control field was found

The control unit is busy to any new operation except sense I/O while performing a scan operation.

A scan found condition is indicated to a test-I/O-and-branch or advance-program-level instruction. The appropriate bit in the status bytes is also set by a scan found condition.

At the end of the operation, the disk control field contains information about the progress of the operation. The identifier portion contains the sector identifier of the last sector scanned unless there is a missing address marker. If there is a missing address marker, the identifier portion indicates the sector with the missing address marker. If no sector was scanned, the identifier portion indicates the first sector designated. The number of sectors scanned can be determined by subtracting the contents of the N-byte from the original value of the N-byte unless all sectors were processed. If all sectors were processed, the N-byte is hex FF.

The 5444/5448 disk data address register contains the original address at the end of the operation unless an equipment check occurs. The contents of the register is unpredictable after an equipment check.

FLAGGING DEFECTIVE 5444/5448 TRACKS

Defective recording areas are handled by track flagging. The flagging procedure included in the disk attachment is used to identify defective tracks and their alternates. Alternate tracks can be assigned under program control at the time a track in cylinders 4-202 is found to be defective. Cylinders 1-3 are provided for assignment as alternate tracks.

The flagging procedure uses bits 6 and 7 of the flag byte of the identifier of each sector recorded on the disk. Bit 6 alone indicates that the track is a defective track; bit 7 alone indicates that the track is an alternate track. Both bits 0 indicates that the track is an original good track. Both bits 1 indicates a defective alternate track (which has its own address in the C-byte when IBM program products are used).

A track with a bad spot is marked defective and an alternate is assigned to replace the whole track. When a track is found to be defective, a write identifier operation must be performed to write the flag bytes with bit 7 = 1 and the C- and S-bytes of the identifiers from the defective track on the alternate track. Then the recoverable data from the defective track must be written on the corresponding sectors on the alternate track. Finally, the defective track must be written with a write identifier operation to write flag bytes with bit 6 = 1 and the C- and S-bytes of the identifiers from the alternate track on the defective track.

Track 203 (used by IBM customer engineers for diagnostics) can be flagged with bit 6 on, bit 7 off if the track is defective. However, the address of an alternate track should not be assigned to track 203 if the track is used for CE diagnostics.

If the flags in storage and on disk are not equal, the attachment sets the track condition check. If a subsequent TIO or APL instruction tests for a not-ready/check condition, (1) the branch occurs for the TIO or (2) the program loops on the APL instruction. (*Note:* the APL should not be issued in Model 15 mode.)

The identifier fields of the tracks are:

- Flag byte of customer usable track

  Good — Bits 6 and 7 of the F-byte are both 0 and the C- and S-bytes contain the cylinder and sector numbers 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 and sector address of the alternate track assigned.

- Flag byte of alternate track

  Good — Bit 6 is 0 and bit 7 is 1 in the F-byte. The C- and S-bytes contain the cylinder and sector addresses from the defective track replaced by the alternate.

  Defective Alternate — Bit 6 is 1 and bit 7 is 1 in the flag byte. Bytes C and S contain the cylinder, head, and sector numbers of the respective sectors.

5444/5448 TRACK INITIALIZATION PROCEDURES

The following procedures must be followed by track initialization programs for the 5444/5448 disk storage drive for System/3. They analyze the condition of the surface and format the tracks.

1. Read identifier to determine that the track has not been previously flagged. This step *must not* be performed when initializing a previously *unused* disk.

2. Write identifier with a data field of hex 55. Write appropriate code into bits 6 and 7 of the flag byte: hex 00 for original tracks, hex 01 for tracks assigned as alternate tracks.

3. Read data of all the sectors to ensure that it can be recovered. If an error occurs, go to step 10.

4. Repeat step 2 with a data field of hex 00.

5. Repeat step 3.

6. Seek to the next track and repeat steps 1 through 5.

7. Repeat steps 1 through 6 until all tracks are processed.

8. Read identifier on all tracks to check for seek errors. If a seek error on the writing operation is detected, initialization must repeat steps 1 through 7 because a seek error on the writing operation causes 2 different tracks to contain the same identifiers or the identifiers for 1 track to be missing.

9. Perform steps 1 through 8 at least once.

10. If an error occurs, the device status must be analyzed. If a missing address marker or data check occurs, retry a read data instruction at least 10 times. On the first unsuccessful retry that indicates missing address marker or 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 marker or data check, follow the normal error recovery procedures.

11. Assign an alternate track unless this is an alternate track.

12. Write identifier on the defective track with the address of the alternate track in the identifier and a value of hex 02 in the flag byte. A defective alternate track should contain its own address and a value of hex 03 in bits 6 and 7 of the flag byte.

13. Set the flag byte in the disk control field to hex 02. Perform a read identifier operation. If the address of the alternate track is not recoverable, the disk must be repaired unless this is an alternate track.

14. Seek to the alternate track.

15. Set the flag byte in the disk control field to hex 01. Write identifier on the alternate track with the identifiers of the defective track in the disk control field. Alternate tracks must be proved reliable by steps 1 through 5 before they are used as alternates.

16. Continue with initialization on the next track.

The basic requirement is for one pass through steps 2 through 8. An option must be provided to allow any number of passes up to 255.

No program should change the flagging of a previously flagged track except as follows:

1. Initialization programs must have the following additional capabilities:
   a. The option to ignore all previously flagged tracks
   b. The option to unconditionally flag or unflag any individual track

2. Operating programs that have provision for dynamic flagging must perform steps 11-15 of this procedure.

SUGGESTED 5444/5448 ERROR RECOVERY PROCEDURES

The following minimum error recovery procedures are defined for the disk and attachment. A test I/O for not-ready/check conditions must be performed. If not-ready/check is present, perform action III from Figure 7-10. The status bytes and bits must be tested in the following order and the actions from Figure 7-10 performed when the bits are set:

Priority	Byte	Bit	Condition	Action
1	0	3	Equipment check	II
2	0	1	Intervention required	VII
3	1	5	Overrun	III
4	0	5	No record found	III
5	0	2	Missing address mark	III
6	0	4	Data check	III
7	0	6	Track condition check	III
8	0	7	Seek check	V
9	1	2	End of cylinder	IV

Action	
I	1. If there is no additional error recovery procedure, perform an operator message and stop. 2. If there is an additional error recovery procedure, exit to it. 3. If the additional error recovery procedure fails, perform an operator message and stop.
II	Retry the original operation or sequence of operations once. On the second occurrence of this error condition, perform an operator message and stop. Upon operator restart, do Action II.
III	1. Perform a read ID operation. If there is an error, do Action VI for the original operation. If no error, update the residual disk drive data register (DDDR) and N-field values. 2. If the present track is defective, indicate an alternate is being used, determine the alternate track from the ID field, and go to Action IV, part 3. 3. Check to determine if head switching from an alternate track has just taken place. If so and no true error exists, go to Action IV, step 2. 4. Go to Action VII.

Action	
IV	1. Update the residual disk address to the next track. 2. Use the residual to obtain the next track address. Set the flag byte to zero. 3. Continue the operation with updated values.
V	1. If 16 recalibrate retries were attempted, go to Action I. 2. Issue a recalibrate. 3. Retry the original operation.
VI	1. If 16 retries have been attempted since the last recalibrate, go to Action V, step 1. 2. Go to Action V, step 3.
VII	Perform an operator message and stop. After restart, repeat the original operation or sequence of operations. **CAUTION:** If a nonrecoverable disk error occurs, have a qualified customer engineer examine both the disk drive and the disk for damage before using the drive or disk for any subsequent disk operations.

Figure 7-10. 5444/5448 Disk Error Recovery Procedures

SUMMARY OF 5444/5448 INSTRUCTION HANDLING

Figure 7-11 summarizes how the system handles disk instructions under various operating conditions.

Instruction Issued to a Disk Drive That:	Is Not Ready	Has an Equipment Check (Not Caused by Unsafe)	Is Unsafe	Has the Ho-op Bit Active	Is Ready and Not Busy	Is Executing a Seek	Is Busy[2]
Start I/O Read Write or Scan	Causes an APL or is rejected[1]	Accepted and executed (resets the equipment check)	No-Op'ed	No-Op'ed	Accepted and executed (brings up FCU busy)	Accepted (will be executed if and when the seek is completed w/o error)	Causes an APL or is rejected[1]
Start I/O Seek	Causes an APL or is rejected[1]	Accepted and executed (resets the equipment check)	No-Op'ed	No-Op'ed	Accepted and executed (brings up seek busy)	Causes an APL or is rejected[1]	Causes an APL or is rejected[1]
Load I/O	Causes an APL or is rejected[1]	Accepted and executed	Accepted and executed	Accepted and executed	Accepted and executed	Accepted and executed	Causes an APL or is rejected[1]
Test I/O							
Error		Branch	Branch	Branch			
Not Ready	Branch		Branch	Branch[3]			
Busy							Branch
Sense I/O	Executed	Executed	Executed	Executed (and resets no-op)	Executed	Executed	Executed

[1] APL for dual program level systems, and rejected and looped on for one program level systems.
[2] FCU busy will become active when, and only when, a read, write, or scan operation is accepted by the FCU (file control unit).
[3] Branch occurs only if no-op was set due to unsafe.

Figure 7-11. Summary of Instruction Handling on 5444/5448

5444/5448 START I/O (SIO)

Unit	Op Code (hex) Byte 1	Q-Byte (binary) Byte 2	R-Byte (binary) Byte 3
5444	F3	101x x xxx	xxxx xxxx
5448	F3	110x x xxx	xxxx xxxx

DA M N Control Code

Bits

N-Code	0123 4567	Function Specified	
000	xxxx xxxx	Seek	
001	xxxx xx00	Read data	
	xxxx xx01	Read identifier	
	xxxx xx10	Read data diagnostic	
	xxxx xx11	Verify	
010	xxxx xxx0	Write data	
	xxxx xxx1	Write identifier	
011	xxxx xx00	Scan equal	
	xxxx xx01	Scan low or equal	
	xxxx xx11	Scan high or equal	
		Model 15	*Models 8 and 10*
100			Invalid N-code; results in processor check.
	1xxx xxxx	Enable interrupts[1]	
	x1xx xxxx	Reset seek 0 interrupt	
	xx1x xxxx	Reset seek 1 interrupt	
	xxxx x1xx	Reset op end interrupt	
	xxxx xx1x	Disable all interrupts	

Any N-code not shown is invalid and causes:
- Program check if interrupt level 7 is enabled on Model 15
- Processor check if interrupt level 7 is not enabled on Model 15
- Processor check on Models 8 and 10

M: 0 specifies the upper disk (removable on the 5444).
1 specifies the lower disk.

DA:
1010 (hex A) specifies 5444 drive 1 as the addressed device.
1011 (hex B) specifies 5444 drive 2 as the addressed device.
1100 (hex C) specifies 5448 drive 1 as the addressed device.
1101 (hex D) specifies 5448 drive 2 as the addressed deivce.

F3 specifies a start I/O operation. F as the first hex character in the op code specifies a command-type instruction (that is, an instruction with no operand addressing).

[1]When N = 100, control code bits 0 and 6 should never be on at the same time.

Operation

The 5444/5448 drive specified by the DA-code performs the function specified by the N-code and control code on the disk specified by the M-code.

Program Notes

- The program loops on the SIO instruction until the condition no longer exists whenever:
  - The program issues an SIO to a control unit that is busy, or
  - The program issues an SIO seek to a drive that is already seeking or is not ready.

- The control unit provisionally accepts a single SIO specifying read, write, or scan for later execution if the addressed drive is executing a seek. If an error occurs during this seek operation, the control unit turns the no-op status bit on without executing the provisionally accepted SIO.

- The program may overlap a seek on one drive with a seek on the other drive.

- The program may overlap a read, write, or scan on one drive with a seek on the other drive if the seek instruction is issued first. Overlap does not occur if the seek is issued during a read, write, or scan operation.

- The SIO instruction uses the contents of the disk data address register as the initial address of all sector data fields. It uses the contents of the disk control address register as the address of the disk control field.

  Unless the SIO specifies an interrupt control function, the control unit sets the no-op status bit and requests an op-end interrupt without executing the instruction if the program issues an SIO to a drive that cannot ensure data integrity because of an unsafe condition (status byte 2, bit 0).

- Executing an SIO resets all previously-generated device status except:
  - Seek check (seek check is also reset if it is associated with the drive specified by the new SIO)
  - Equipment check caused by an unsafe condition
  - Cylinder zero (cylinder zero resets when the access arm moves away from that cylinder)
  - No-op
  - Intervention required

Op End Interrupt (Model 15)

The 5444 attachment presents an op end interrupt request to the processing unit at the end of the processing unit instruction during which one of the following conditions occurred on the selected drive:

- The drive completed a data transfer operation (either read or write).

- The drive finished a seek operation.

- The SIO was no-oped because of an equipment check.

5444/5448 LOAD I/O (LIO)

Unit	Op Code (hex)	Q-Byte (binary)	Operand Address	
	Byte 1	Byte 2	Byte 3	Byte 4
5444	31	101x 0 xxx	Operand 1 address	
	71	101x 0 xxx	Op 1 disp from XR1	
	B1	101x 0 xxx	Op 1 disp from XR2	
5448	31	110x 0 xxx	Operand 1 address	
	71	110x 0 xxx	Op 1 disp from XR1	
	B1	110x 0 xxx	Op 1 disp from XR2	

DA M N

N-Code	To Be Loaded
011	CE diagnostic data
100	Disk data address register
110	Disk control address register

Any N-code not shown is invalid and causes:

- Program check if interrupt level 7 is enables on Model 15
- Processor check if interrupt level 7 is not enabled on Model 15
- Processor check on Models 8 and 10

M: Not used in this instruction; should be 0.

DA:
1010 (hex A) specifies 5444 drive 1 as the addressed device.
1011 (hex B) specifies 5444 drive 2 as the addressed device.
1100 (hex C) specifies 5448 drive 1 as the addressed device.
1101 (hex D) specifies 5448 drive 2 as the addressed deivce.

31, 71, or B1 specifies a load I/O operation. The first hex character in the op code specifies the type of operand addressing for the instruction.

Operation

The processing unit loads the 2 bytes of data contained in the operand into the register specified by the N-code. The operand is addressed by its low-order (higher numbered) storage position. If the 5444/5448 no-op bit is on, the processing unit bypasses this instruction and immediately accesses the next sequential instruction. If the addressed register is busy, the program loops on the instruction until the register becomes not busy.

Program Notes

- LIO does not set any 5444/5448 status conditions.
- The processing unit executes the LIO instruction while the addressed drive is executing a seek or recalibrate operation if a read, write, or scan was not accepted or provisionally accepted.
- The processing unit executes the LIO instruction without consideration of the addressed drive ready status.

5444/5448 TEST I/O AND BRANCH (TIO)

Unit	Op Code (hex) Byte 1	Q-Byte (binary) Byte 2	Operand Address Byte 3	Byte 4
5444	C1	101x x xxx	Operand 1 address	
	D1	101x x xxx	Op 1 disp from XR1	
	E1	101x x xxx	Op 1 disp from XR2	
5448	C1	110x x xxx	Operand 1 address	
	D1	110x x xxx	Op 1 disp from XR1	
	E1	110x x xxx	Op 1 disp from XR2	

DA M N

N-Code	Condition Tested
000	Not ready/check. This condition indicates that the drive needs operator intervention or that one of the following device conditions has occurred on one of the drives. To determine the cause of the not-ready/check status, issue a SNS instruction testing the unit for: Data check; No record found; Track condition check; Equipment check not caused by unsafe; Missing address marker; No-op; End of cylinder; Overrun
010	Busy. This condition indicates that the disk drive control unit is executing a read, write, or scan operation or has provisionally accepted such an operation.
100	Scan found. Indicates that one of the drives has been addressed and a scan has been matched in one of the drives. Issue an SNS instruction to determine which drive contained the scan-found condition. The scan-found indication is reset by the next SIO instruction.

Any N-code not shown is invalid and causes:
- Program check if interrupt level 7 is enabled on Model 15
- Processor check if interrupt level 7 is not enabled on Model 15
- Processor check on Models 8 and 10

M:
0 = Test for all conditions except Model A3 not ready on the 5444. Test for all conditions on the 5448.
1 = Test for Model A3 not ready condition if N = 000 (5444 only). Not used on the 5448.

DA:
1010 (hex A) specifies 5444 drive 1 as the addressed device.
1011 (hex B) specifies 5444 drive 2 as the addressed device.
1100 (hex C) specifies 5448 drive 1 as the addressed device.
1101 (hex D) specifies 5448 drive 2 as the addressed deivce.

C1, D1, or E1 specifies a test I/O branch operation. The first hex character in the op code specifies the type of operand addressing for the instruction.

Operation

The control unit tests the drive specified by the DA-code for the conditions specified by the M-code and the N-code. If any of the tested conditions exists, the program branches to the operand address. If no tested condition exists, the program proceeds with the next sequential instruction.

Program Note

An error is indicated if a seek check or unsafe condition is detected for the drive addressed. A seek check or unsafe condition for the other drive does not cause an error to be indicated. To determine which drive caused the check, examine the status data for the drive address.

Resulting Condition Register Setting

This instruction does not affect the condition register.

5444/5448 ADVANCE PROGRAM LEVEL (APL)

Unit	Op Code (hex) Byte 1	Q-Byte (binary) Byte 2	R-Byte (binary) Byte 3
5444	F1	101x x xxx	0000 0000
5448	F1	110x x xxx	0000 0000

DA M N

R-byte is not used in an APL instruction.

N-Code	Condition Tested
000	Not ready/check. This condition indicates that the drive needs operator intervention or that one of the following device conditions occurred on one of the drives. To determine the cause of the not-ready/check status, issue a SNS instruction testing the unit for: Data check, No record found, Track condition check, Equipment check not caused by unsafe, Missing address marker, No-op, End of cylinder, Overrun
010	Busy. This condition indicates that the disk drive control unit is executing a read, write, or scan operation or has provisionally accepted such an operation.
100	Scan found. Indicates that one of the drives has been addressed and a scan has been matched in one of the drives. Issue a SNS instruction to determine which drive contained the scan-found condition. The scan-found indication is reset by the next SIO instruction.

Any N-code not shown is invalid and causes:
- Program check if interrupt level 7 is enabled on Model 15
- Processor check if interrupt level 7 is not enabled on Model 15
- Processor check on Models 8 and 10

M:
0 = Test for all conditions except Model A3 not ready on the 5444.
Test for all conditions on the 5448.
1 = Test for Model A3 not ready condition if N = 000 (5444 only).
Not used on the 5448.

DA:
1010 (hex A) specifies 5444 drive 1 as the addressed device.
1011 (hex B) specifies 5444 drive 2 as the addressed device.
1100 (hex C) specifies 5448 drive 1 as the addressed device.
1101 (hex D) specifies 5448 drive 2 as the addressed deivce.

F1 specifies an APL operation. F as the first hex character in the op code identifies a command-type instruction (that is, an instruction without operand addressing).

Operation

This instruction tests for the conditions specified in the Q-byte.

- Condition present:
  - Systems with Dual Program Feature installed and enabled, activate the inactive program level.
  - Systems without Dual Program Feature installed or with Dual Program Feature installed but not enabled, loop on the advance program level instruction until the condition no longer exists.
- Condition not present: Systems with or without Dual Program Feature access the next sequential instruction in the active program level.

Program Notes

- An error is indicated if there is a seek check or unsafe condition for the drive addressed. A seek check or unsafe condition for the other drive does not cause an error to be indicated. To determine which drive caused the check, examine the status data for the drive address.
- For additional information concerning the advance program level instruction, see Chapter 2.

5444/5448 SENSE I/O (SNS)

Unit	Op Code (hex) Byte 1	Q-Byte (binary) Byte 2	Operand Address Byte 3	Byte 4
5444	30	101x x xxx	Operand 1 address	
	70	101x x xxx	Op 1 disp from XR1	
	B0	101x x xxx	Op 1 disp from XR2	
5448	30	110x x xxx	Operand 1 address	
	70	110x x xxx	Op 1 disp from XR1	
	B0	110x x xxx	Op 1 disp from XR2	

DA M N

N-Code — Sensed Unit

N-Code	Sensed Unit
010	Status bytes 0 and 1[1]
011	Status bytes 2 and 3[1]
100	Disk read/write address register
110	Disk control address register

Any N-code not shown is invalid and causes:

- Program check if interrupt level 7 is enabled on Model 15
- Processor check if interrupt level 7 is not enabled on Model 15
- Processor check on Models 8 and 10

M: Not used; should be 0.

DA:

1010 (hex A) specifies 5444 drive 1 as the addressed device.
1011 (hex B) specifies 5444 drive 2 as the addressed device.
1100 (hex C) specifies 5448 drive 1 as the addressed device.
1101 (hex D) specifies 5448 drive 2 as the addressed deivce.

30, 70, or B0 specifies a sense I/O operation. The first hex character in the op code specifies the type of operand addressing for the instruction.

[1]Odd-numbered status bytes are transferred to the rightmost (higher-numbered) position of the operand. Even numbered status bytes are transferred to the leftmost position of the operand.

Operation

The 5444/5448 attachment transfers 2 bytes of status information or the contents of a register (as specified by the N-code) to the operand. Information transferred applies to the drive specified by the DA-code. The attachment accepts the SNS instruction even though other operations may be in progress. Status bits are defined in Figure 7-12.

Program Note

The following indications are sent only with the status byte associated with the drive being tested:

Equipment check caused by unsafe
Cylinder 0
Seek check
Seek busy
Intervention required
Unsafe
Head settling
Index

All the status bits not listed are returned with the status bytes to a sense I/O for either drive. All status bits except no-op are reset by the next start I/O instruction issued to either drive. No-op is reset by the sense I/O instruction to either drive that transfers it to storage.

Byte	Bit	Name	Indicates	Reset By
0	0	No-op	Last disk instruction was not executed because (1) the selected disk was unsafe (see *Byte 2, Bit 0*) or (2) a check condition occurred during a seek on a drive that has provisionally accepted a read, write, or scan instruction.	Check reset, system reset, or the sense I/O that transfers the no-op bit to storage
0	1	Intervention required[1]	Addressed drive is not ready. Addressing drive 2 in a system with only one drive or addressing the fixed disk on drive 2 when only the removable disk is installed also sets this bit. *Note:* Ready may have dropped because an access reverse operation specified more tracks than the actual number of tracks from the current track to the home position.	Correcting the condition causing the indication If the drive is not ready and no permanent error exists, stop the drive and then restart it to establish ready.
0	2	Missing address mark	This bit indicates a sector number is missing from a sector, sectors are numbered out of sequence, or an address marker is missing making it impossible for the drive to read the sector identification number. This bit is not set if a data check is detected in a sector ID field read to ensure sector sequence.	Next SIO operation or system reset
0	3	Equipment check[1]	The attachment feature detected a hardware failure or cannot guarantee integrity of data being read from the selected drive.	Next SIO operation or 5444 reset if check caused by unsafe equipment
0	4	Data check	The attachment feature detected a read error in an ID or data field of a sector.	Next SIO operation or system reset
0	5	No record found	The sector specified for a read, write, verify, or scan operation was not found on the currently active track. This error may have occurred because the previous operation caused head switching to head 1 while the current operation referred to head 0.	Next SIO operation or system reset
0	6	Track condition check	Bits 6 and 7 of the flag byte on the track do not match bits 6 and 7 of the flag byte in the disk control field. This indicates a defective track is being read and the program should seek the specified alternate track (see *Flagging Defective 5444/5448 Tracks* in this section).	Next SIO operation or system reset
0	7	Seek check[1]	The program specified a cylinder outside the capacity of the disks installed or the attachment feature detected a seek error.	Next SIO addressed to the affected drive or system reset
1	0	Scan equal hit	The equal condition was satisfied during a scan operation.	Next SIO or system reset

[1] Applies only to the drive addressed

Figure 7-12 (Part 1 of 3). 5444/5448 Disk Drive Status Bytes

Byte	Bit	Name	Indicates	Reset By
1	1	Cylinder 0[1]	The selected drive's read head is positioned at cylinder 0.	Read head moving from cylinder 0
1	2	End of cylinder	One of the following has occurred on a multiple sector or scan operation: ● The last sector on the disk (sector 55) was operated on and the number of sectors specified for the operation still has not been satisfied. That is, the instruction attempted to operate beyond the end of the cylinder. ● Head 1 IDs were written on the upper surface of the disk (to identify lower tracks on the upper surface as alternate tracks) and the instruction tried to operate beyond the end of the track (head switching). All sectors through the last one on the cylinder were successfully operated upon when this check occurred.	Next SIO executed or system reset
1	3	Seek busy[1]	The drive addressed by the SNS instruction is performing a seek operation.	End of seek operation or system reset
1	4	5444 Model 1/A1 N/A to 5448	A 5444 Model 1 or Model A1 is installed on the system. This bit is always inactive on the 5448.	Taking the 5444 Model 1 or Model A1 off the system
1	5	Overrun	Processing unit did not allow a cycle steal to the 5444/5448 in time to transfer data before it was lost. This occurs during a processor check stop while the processing unit clock is not running or if operating higher priority devices.	Processing unit restart procedure
1	6	Status address A	These two bits specify the drive that was addressed for the last read, write, or scan operation. Attachment dependent status bits apply to the drive identified by these bits: Bits 6 and 7 = 00 specifies drive 1 Bits 6 and 7 = 01 specifies drive 2	Next 5444/5448 sense command or system reset
2	0	Unsafe[1]	A 5444/5448 condition that could cause faulty reading or writing exists. To determine the cause of the unsafe condition, interrogate the other bit positions in this status byte.	Removing the condition that turned the unsafe bit on, 5444/5448 reset
2 2 2	1 2 3	Timing analysis program (TAP) lines A, B, C (CE diagnostic bits)	Normally used by the CE to further define unsafe condition.	
2	4	Index[1]	Index area of the disk is passing under the read head. This bit is turned on when the area starts under the read head. The bit remains on for about 43 μs, the approximate length of time required for the entire index area to pass under the read head.	Trailing edge of the index area passing the read head

[1] Applies only to the drive addressed

Figure 7-12 (Part 2 of 3). 5444/5448 Disk Drive Status Bytes

Byte	Bit	Name	Indicates	Reset By
2	5	Head setting[1]	The seek operation is not complete because the head is still moving slightly.	All head movement and vibration stopping
2	6	CE sense bit	Various control unit conditions. The CE uses these indications for diagnostic programming.	CE action
2	7	Not used		
3	0	CE sense bit	Various control unit conditions. The CE uses these indications for diagnostic programming.	CE action
3	1	CE sense bit		
3	2	CE sense bit		
3	3	Seek 0 complete[2]	The seek last initiated on drive 1 has ended.	SIO reset interrupt or system reset
3	4	Seek 1 complete[2]	The seek last initiated on drive 2 has ended.	
3	5	Op-end[2]	An operation being performed on the 5444 has ended (see *5444/5448 Start I/O (SIO)* in this section).	SIO reset interrupt or system reset
3	6	CE sense bit	Diagnostic indications. The CE uses these indications for diagnostic programming.	CE action
3	7	CE sense bit		

[1] Applies only to the drive addressed
[2] Applies only to Model 15. Not used on the 5448.

Figure 7-12 (Part 3 of 3). 5444/5448 Disk Drive Status Bytes

IBM 5445 Disk Storage

The IBM 5445 provides large capacity, high speed, direct access storage capability for System/3. The 5445 is available in three models. Models 1 and 2 each contain the mechanism to drive one removable IBM 2316 Disk Pack (Figure 7-13), Model 1 contains the power supply for itself and a Model 2, and must be attached to a storage control special feature within the processing unit. Model 2 must be attached to the Model 1. Model 3 is a combination of the Model 1 and Model 2 housed within a single set of covers and equipped with a single power supply. The unit must be attached to the same storage control feature in the 5415 as the Model 1. The 5445 is mutually exclusive with the IBM 5448 Disk Storage Drive.

IBM 2316 DISK PACK

The IBM 2316 Disk Pack (Figure 7-13) is a compact disk assembly, 15 inches in diameter (with cover), and weighs about 13 pounds. The disk pack contains 11 disks, each 14 inches in diameter. Disks are mounted one-half inch apart on a vertical shaft. The disks provide 20 surfaces on which data can be recorded (the top of the upper disk and the bottom of the lower disk are not used). The entire assembly rotates once every 25 milliseconds.

Care and handling procedures for 2316 Disk Packs are described in *IBM Disk Pack and Cartridge Handling Procedures,* GA26-5756.

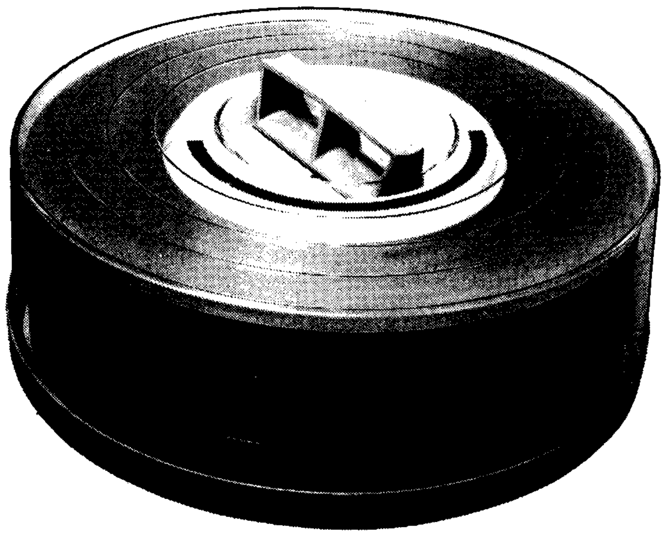


Figure 7-13. IBM 2316 Disk Pack

5445 PHYSICAL CHARACTERISTICS

Valid configuration of 5445s per system:

- One Model 1 (20.48 million bytes total)
- One Model 1 and one Model 2 (40.96 million bytes total)
- One Model 3 (40.96 million bytes total)
- Two Model 1's and one Model 2 (61.44 million bytes total)
- Two Model 1's and two Model 2's (81.92 million bytes total)
- Two Model 3's (81.92 million bytes total)
- One Model 1 and one Model 3 (61.44 million bytes total)
- One Model 1, one Model 2, and one Model 3 (81.92 million bytes total)

Data rate	312 kilobytes/second
Disk rotation speed	2400 rpm
Average rotational delay	12.5 milliseconds
Maximum access time	130 milliseconds
Average random access time	60 milliseconds
Minimum access time (single track movement)	25 milliseconds
Capacity per drive	20.48 megabytes
Number of data cylinders[1]	
Model 1	200
Model 2	200
Model 3	400
Number of alternate (spare) cylinders[1]	
Model 1	3
Model 2	3
Model 3	6
Data tracks per cylinder	20
Number of maximum-size data records per track[2]	20
Capacity per physical record (maximum)	256 bytes (key and data)

[1] As used with IBM programming systems support.
[2] Programming logical records can be one or more physical records long.

5445 ACCESS MECHANISM AND DISK ORGANIZATION

The 5445 reads information from and writes information onto disk surfaces of the 2316 disk pack by means of read/write heads. A movable access mechanism positions 20 read/write heads under control of the 5445 attachment feature, which responds to seek instructions issued from the program. Each access arm (similar to a tooth in the comb) holds two heads: one on the top of the arm, and one on the bottom. Heads are numbered, from top to bottom of the access mechanism, from 0 through 19. Therefore, heads 0 and 1 are attached to the upper arm, 2 and 3 to the second arm, etc. While the drive is operating, a cushion of air holds each head off the disk surface.

The 20 read/write heads always occupy a common vertical plane; that is, all 20 heads are always aligned one above the other, so that any movement of the access mechanism causes identical movement of all heads. Therefore, 20 different tracks—one for each of the 20 disk surfaces used—are always under the read/write heads (one head for each track) at each access arm position. This means that 20 tracks are available for read/write operations without moving the access mechanism. (Heads are numbered 0 through 19, to identify the disk surface each head reads from and writes onto.)

Figure 7-14 shows how the entire disk pack constitutes 203 concentric cylinders of information. Cylinder numbering is from 000 (outermost cylinder) to 202. Tracks in cylinders 200, 201, and 202 are specified by IBM programming support as alternate tracks, and tracks in cylinders 000 through 199 as primary tracks. If one of the primary tracks is defective, the program assigns an alternate track to replace the defective track.

Each unique track has an address that consists of the track cylinder number followed by its read/write head number.

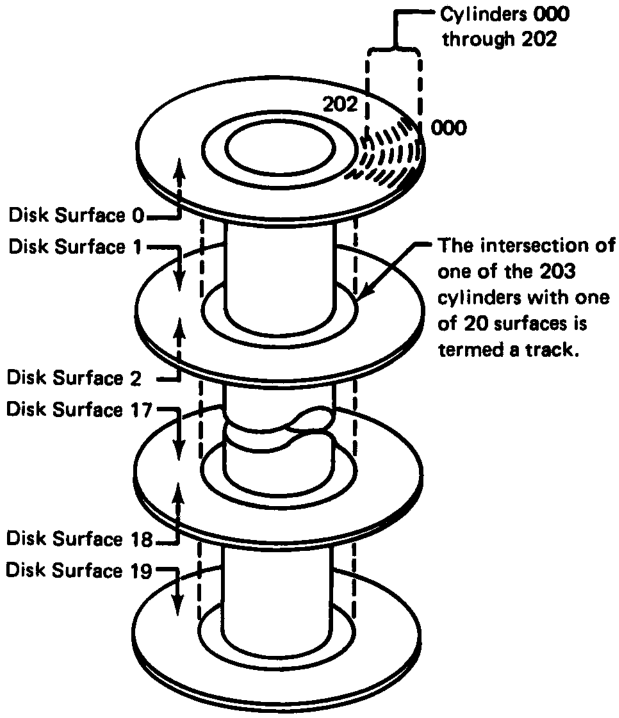


Figure 7-14. 5445 Cylinder Concept

5445 DATA COMPATIBILITY

Data written on a 2316 disk pack by the 5445 can be read by any IBM 2314 or 2319 Disk Storage Drive; data recorded by a 2314 or 2319 can be read by a 5445 if the records are formatted using the formatting procedures specified for the 5445. When such formatting is followed, the 2316 disk packs provide data interchangeability between the IBM System/3, IBM System/360, and IBM System/370.

5445 DATA FORMAT

Data is recorded on the 2316 disk pack in variable record length format, with a maximum length of 256 bytes for the combined key and data fields. Twenty maximum length records can be recorded on 1 track. Decreasing the record length increases the number of records that can be recorded on a track. (When IBM programming support is used, the key length is always 0, and the data length is always 256; however, record lengths may span physical records.)

5445 TRACK FORMAT

Figure 7-15 shows disk organization and track format. The format for each track written on the disk pack starts at a point on the disk called the index marker. (This point is specified by a signal emitted by the disk spindle as it turns all the disks, generating synchronized index markers for all tracks on all disks in the pack.) A home address, then record 0 (a track identifier record), then sequentially numbered records follow the index marker on the track until the index marker is again encountered, signalling that the entire track was used. Gaps, automatically written by the attachment, separate the various unique format units and areas in the records.

5445 Index Marker △

The index marker signals the initial point of each track. The index marker is not recorded on the track or in storage. However, it is shown in figures in this manual as a track reference point.

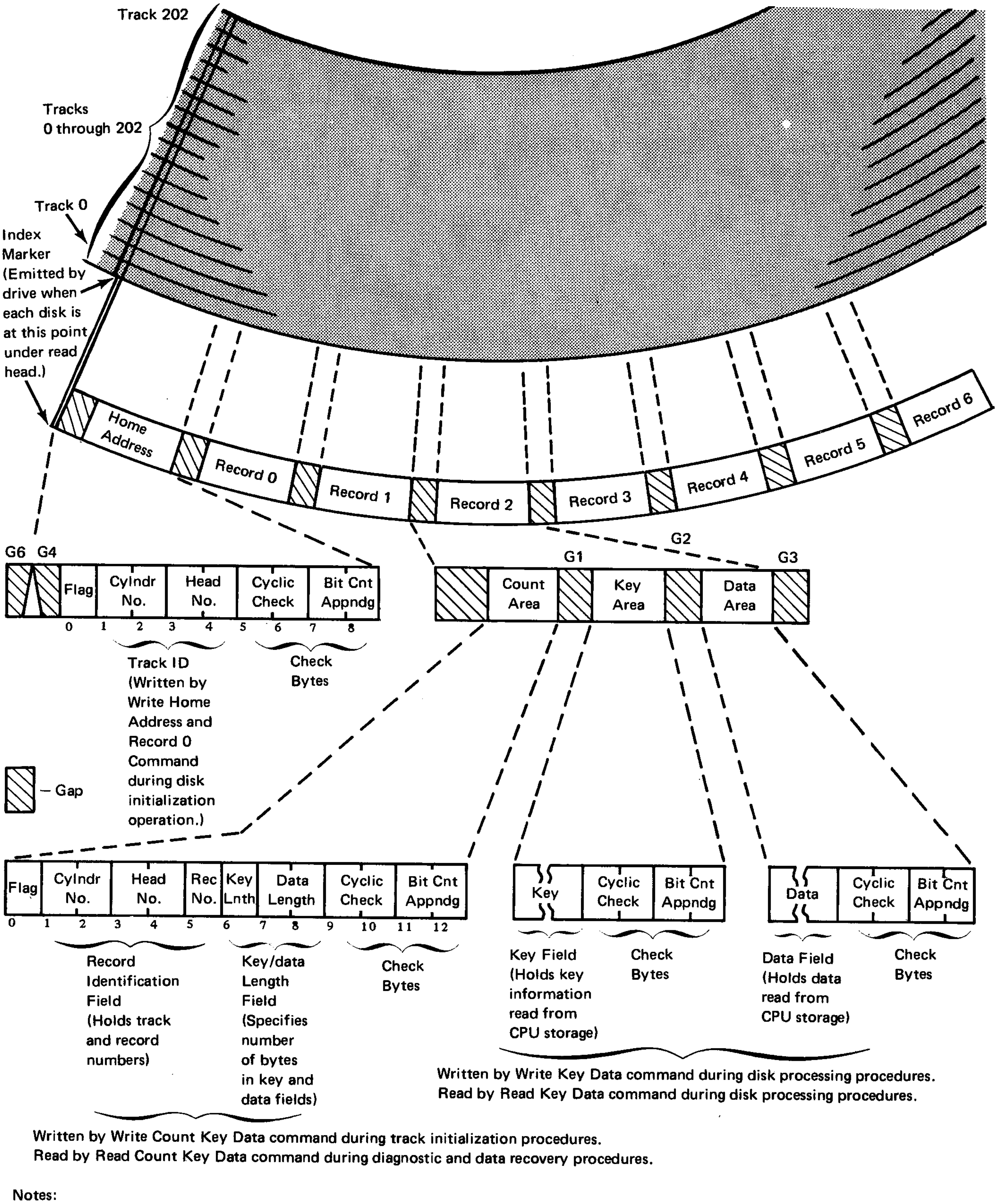
5445 Gap

A gap is an area written on the track by the attachment to separate two adjacent groups of data and to identify the group that follows the gap. This information is used by the attachment only.

5445 Home Address (HA)

Flag	Cylinder Number	Head Number	Cyclic Check	Bit Count Appendage
0	1 2	3 4	5 6	7 8

The home address gives each track a unique track identity that is not affected by normal programming operations. Each track in a storage drive can be located directly by cylinder number and head number. Normal programming operations can use the home address area without changing its contents. Home addresses are transferred from the processing unit to the 5445 by a write home address command, and from the 5445 to the processing unit by a read home address command. Home addresses are usually written by utility programs during file initialization.



Notes:

1. Records are numbered consecutively from zero for correct machine operation.
2. Key and data fields are variable length; therefore, track formats are not identical in record locations. If key length of zero is specified, the key field and its preceding gap are not included in format.

Figure 7-15. 5445 Track Format and Disk Layout

5445 Home Address Flag Byte (F)

Flag	Cylinder Number	Head Number	Cyclic Check	Bit Count Appendage
0	1 2	3 4	5 6	7 8

The home address flag byte indicates track condition and whether the track is a primary track or an alternate track. The flag byte can be transferred to the CPU by a read home address command.

Normally, all 8 bits of the flag byte are 0 when the home address is first written by a write home address command. Thereafter, the flag bits assume significance:

Bit	State	Meaning
0	0 or 1	Internal control bit
1	0 or 1	Special control bit in write HA and R0 operation
2	—	Not used
3	—	Not used
4	—	Not used
5	—	Not used
6	0	Track is operative
	1	Track is defective
7	0	Track is a primary track
	1	Track is an alternate track

Bits 6 and 7 must be program-propagated into the flag byte of each record on the track; otherwise, a check occurs.

5445 Home Address Cylinder Number Bytes (CC)

Flag	**Cylinder Number**	Head Number	Cyclic Check	Bit Count Appendage
0	1 2	3 4	5 6	7 8

The group of tracks available to the 20 read/write heads at each access mechanism position comprise a cylinder. The cylinder number identifies the cylinder within which the track is situated. All bits in the first byte must be 0; the next byte holds the cylinder number (000 through 202 decimal, or 00 through CA hex).

5445 Home Address Head Number Bytes (HH)

Flag	Cylinder Number	**Head Number**	Cyclic Check	Bit Count Appendage
0	1 2	3 4	5 6	7 8

This 2-byte field identifies the read/write head associated with the specified track. The head number, together with the cylinder number, identify a single track to be acted upon. The bits in the first head-number byte all must be 0's; the second byte must contain the head number (00 through 19 decimal, or 00 through 13 hex).

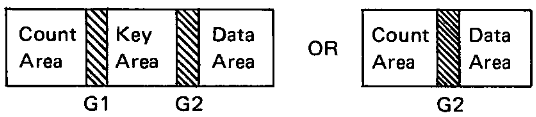
Note: The disk module holding the disk pack is specified by the M-bit in the program instruction used to initiate the I/O operation.

5445 Home Address Cyclic Check and Bit Count Appendage Bytes (Check Bytes)

Flag	Cylinder Number	Head Number	**Cyclic Check**	Bit Count Appendage
0	1 2	3 4	5 6	7 8

The 2 cyclic check and 2 bit count appendage bytes are generated by the attachment and used by the attachment for error detection and recovery. The leftmost bit count appendage byte is called the BCI byte, and indicates which disk drive wrote the record: hex C1 = 5445 drive 1, hex C2 = 5445 drive 2, hex C3 = drive 3, hex C0 = drive 4.

5445 Records (R0, R1, R2, etc)



Records, consecutively numbered from record 0 (R0) upward, fill the track from the home address to the end of the track (detected by encountering the index marker).

Each record contains a count area and either (1) a data area only or (2) both a key area and a data area. The number of records that can be formatted on a track is a function of the assigned lengths of the key areas and data areas for the records being formatted.

5445 Record Count Area

Count Area	Key Area	Data Area

Flag	Cylinder Number	Head Number	Record Number	Key Length	Data Length	Cyclic Check	Bit Count Appendage
0	1 2	3 4	5	6	7 8	9 10	11 12

The count area identifies the record and defines the number of bytes in the key and data areas of the record. During record operations, the attachment compares the record identification data (cylinder number, head number, and record number) in the disk drive control field in processing unit storage with the cylinder number, head number, and record number bytes in the count area of records passing under the read head. A compare equal condition indicates that the desired record is under the read head: this is called *record orientation.* If no orientation occurs, the attachment posts a no-record-found indication.

5445 Record Count Area Flag Byte (F)

Flag	Cylinder Number	Head Number	Record Number	Key Length	Data Length	Cyclic Check	Bit Count Appendage
0	1 2	3 4	5	6	7 8	9 10	11 12

The record count area flag byte is formatted by the 5445 attachment from information stored in the disk drive control field in the processing unit. Flag bit significance and their settings are:

Bit	State	Meaning
0	0	Indicates even-numbered record
	1	Indicates odd-numbered record
1	–	Not used; should be 0
2	–	Not used; should be 0
3	–	Not used; should be 0
4	–	Not used; should be 0
5	–	Not used; should be 0
6	0	Indicates operative track
	1	Indicates defective track
7	0	Indicates track is a primary track
	1	Indicates track is an alternate track

The attachment causes bits 6 and 7 for all records on the track to be set to the values of the corresponding bits in the home address flag byte.

5445 Record Count Area Cylinder Number Bytes (CC)

Flag	**Cylinder Number**	Head Number	Record Number	Key Length	Data Length	Cyclic Check	Bit Count Appendage
0	1 2	3 4	5	6	7 8	9 10	11 12

The cylinder number identifies the cylinder within which the record is stored. All bits in the first byte must be 0; the next byte holds the cylinder number, which is assigned by the program. The cylinder number is written from the disk drive control field during a write count key data operation; it is not checked by the 5445.

5445 Record Count Area Head Number Bytes (HH)

Flag	Cylinder Number	**Head Number**	Record Number	Key Length	Data Length	Cyclic Check	Bit Count Appendage
0	1 2	3 4	5	6	7 8	9 10	11 12

The 2-byte field identifies the read/write head associated with the track on which the record is to be placed or from which the record is to be read. The head number, together with the cylinder number, identifies the track associated with the record. The bits in the leftmost head-number byte must be all 0's; the second head-number byte holds the head number, which is assigned by the program. The head number is written from the disk drive control field during a write count key data operation; it is not checked by the 5445.

5445 Record Count Area Record Number Byte (R)

Flag	Cylinder Number	Head Number	**Record Number**	Key Length	Data Length	Cyclic Check	Bit Count Appendage
0	1 2	3 4	5	6	7 8	9 10	11 12

This byte identifies a particular record on the specified track. Records are numbered sequentially on the track, starting with the number assigned by the program to record 0. The number assigned to record 0 is not checked by the 5445, but must be hex 00 for correct disk drive operation. The number of records per track is limited by the addressing capability and by the track capacity. (If the read/write head encounters the index marker point on the disk track during write count key data operation, the track capacity was exceeded.) The record number is written on the track from the disk drive control field during a write count key data operation.

5445 Record Count Area Key Length Byte (KL)

Flag	Cylinder Number	Head Number	Record Number	**Key Length**	Data Length	Cyclic Check	Bit Count Appendage
0	1 2	3 4	5	6	7 8	9 10	11 12

The key length byte specifies the number of bytes in the key area of the record (excluding the cyclic check and bit count appendage bytes, which are check bytes). Valid key lengths are 0 through 255 decimal, or 0 through FF hex. However, in System/3 the key length is also conditioned by the data length specified, because the total value of the key area plus the data area on the record cannot exceed 256 bytes (decimal). For those installations using IBM programming support, the key length must be 0.

5445 Record Count Area Data Length Bytes (DL)

Flag	Cylinder Number	Head Number	Record Number	Key Length	**Data Length**	Cyclic Check	Bit Count Appendage
0	1 2	3 4	5	6	7 8	9 10	11 12

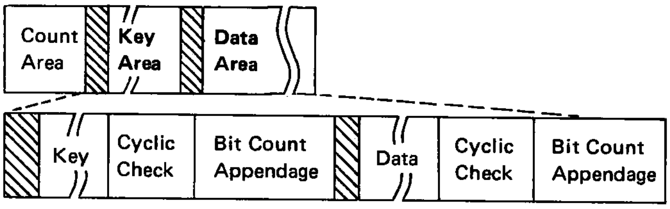
The data length bytes specify the number of bytes in the data area of the record (excluding the cyclic check and bit count appendage bytes, which are check bytes). Valid data lengths are 0 through 256 decimal, or 0 through 100 hex. However in System/3, the data length is also conditioned by the key length specified, because the total value of the data area plus the key area on the record cannot exceed 256 bytes (decimal). For those installations using IBM programming support, the data length must be 256 bytes for all records except record 0, which is assigned a data length of 8 bytes. Note that the last position of a specified data field must not be the byte that immediately precedes the last byte location in storage. This results in an invalid address processor check when the DDDR is set to its final value.

5445 Record Count Area Cyclic Check and Bit Count Bytes

Flag	Cylinder Number	Head Number	Record Number	Key Length	Data Length	**Cyclic Check**	Bit Count Appendage
0	1 2	3 4	5	6	7 8	9 10	11 12

The 2 cyclic check and 2 bit count appendage bytes are generated by the attachment and used by the attachment for error detection and recovery. The leftmost bit count appendage byte is called the BCI byte, and indicates which disk drive wrote the record: hex C1 = 5445 drive 1, hex C2 = 5445 drive 2, hex C3 = 5445 drive 3, hex C0 = 5445 drive 4.

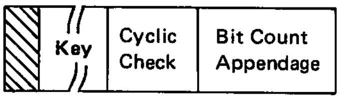
5445 Record Key Area and Data Area (Key/Data Area)



These two record format areas hold the application-oriented information in the record, and should always be considered as a single entity for System/3 programming and operations. For example, the total number of bytes of combined key and data information in a record cannot exceed 256. If the key area is omitted from the record (a key length byte of 0), the gap preceding the missing area is also omitted from the record.

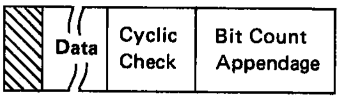
Note: When counting data bytes for the areas, do not consider the cyclic check and bit count appendage bytes as part of the areas.

5445 Record Key Area Key Bytes



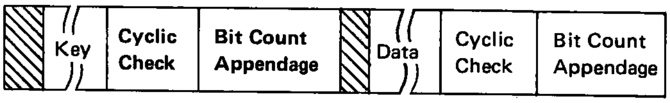
The key-area key bytes can contain record identifying information such as serial number, social security number, or policy number. The number of key bytes in the key area is specified by the key-length byte in the count area, but can never exceed 255.

5445 Record Data Area Data Bytes



The data-area data bytes can contain the information identified by the count and key areas of the record. Data information is organized and arranged by the programmer. The number of data bytes in the data area is specified by the data-length bytes in the count area, but can never exceed 256.

5445 Record Key/Data Area Cyclic Check and Bit Count Bytes

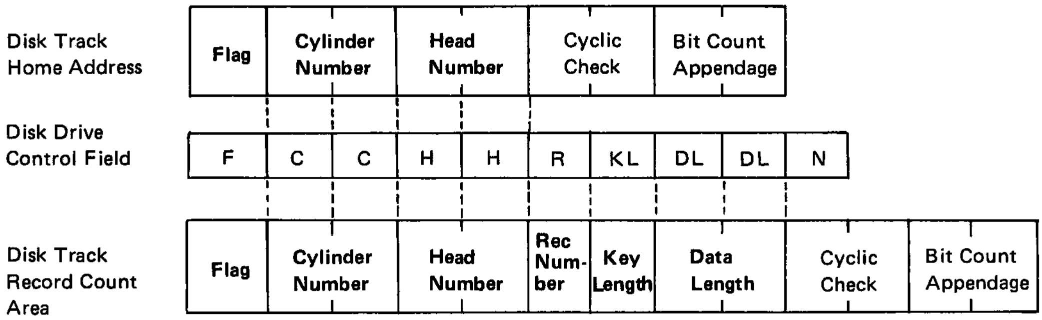


The cyclic-check and bit-count-appendage bytes are generated by and used by the attachment for error detection and recovery. The leftmost byte of each set of bit count appendage bytes is called a BCI (bit count indicator) byte, and indicates which disk drive wrote the record: hex C1 = 5445 drive 1, hex C2 = 5445 drive 2, hex C3 = drive 3, hex C0 = drive 4.

5445 DISK DRIVE CONTROL FIELD (DDCF)

F	C	C	H	H	R	KL	DL	DL	N

The disk drive control field is a program-defined field in main storage that contains a 10-byte control argument for all start I/O instructions. The DDCF can start on any byte boundary addressed by the disk drive control register (DDCR). As shown below, all the bytes except the N-byte in the DDCF have directly related bytes in the disk home address and record count areas:



It is generally necessary to preload the defined DDCF with the control argument for the operation before issuing a disk-related start I/O command. Program modification of the DDCF must not be attempted while the disk drive attachment is busy. The functional significance of each DDCF byte except the R-byte and the N-byte is identical to that of the corresponding byte in the disk track record count area.

5445 DDCF R-Byte

F	C	C	H	H	**R**	KL	DL	DL	N

This byte specifies the sequential number of the record on the track. Valid record numbers are 0 through 255 decimal (00 through FF hex). The R-byte must match the corresponding byte in the disk count area before record orientation can occur.

5445 DDCF N-Byte

F	C	C	H	H	R	KL	DL	DL	**N**

This byte specifies the number of additional fixed format records to be operated on. Therefore, a control field with an N-byte of hex 5 specifies an operation on the addressed record and the following five records.

5445 Multiple Fixed Format Records (Multiple Records)

Fixed format records are defined as contiguous records having equal length key areas and equal length data areas. Therefore, a control field with an N-byte specifying other than 0 causes a multiple fixed format record (often referred to simply as multiple record) operation.

After a record has been successfully operated on, the attachment:

1. Decrements the N-byte by 1 (decrementing by 1 from an N-byte value of 0 places a hex FF in the N-byte).

2. Examines the contents of the N-byte. If the N-byte holds FF, there are no more records to be operated on and the operation ends. If the N-byte contains other than FF, the value contained in the N-byte, plus 1, specifies how many more records must be operated on, and the attachment continues with step 3.

3. Increments the DDCF R-byte by 1 to specify the next sequential record as the record to be operated on.

4. Performs the operation specified by the instruction on the record specified by the updated R-byte.

Note: These functions are slightly modified if head switching occurs during the operation.

5445 HEAD SWITCHING

During multiple-record operations, a single start I/O instruction can cause as many as 256 records to be operated upon (the record specified by the R-byte, plus another 255 identically formatted records trailing the specified record in the disk drive, as specified by an N-byte of hex FF in the DDCF). In many cases, some of the multiple records must be read from the originally specified track, and the next records must be read from a second track. To operate on records from 2 different tracks as the result of a single instruction, the attachment switches read/write heads switching from the presently active head to the next higher numbered head after the last record on the original track has been operated upon. During head-switching, the attachment increments the head number by 1 and resets the record number to 1. Therefore, the next record operated on is record 1 (note that record 0 is bypassed) of the newly selected track.

Note: If head switching occurs from head 19 to head 20 (which is a nonexistent head) the file stops with an end-of-cylinder condition posted.

5445 RESIDUAL VALUES

The data held by the DDCF, DDCR, DDDF, and DDDR at the end of each start I/O operation is particularly important for error recovery. These residual values at the end of each normal-end I/O operation are discussed with the write-up about the operation. This section defines residual values when check ending status posted.

5445 Disk Drive Control Field (DDCF) Residuals

F	C	C	H	**H**	**R**	KL	DL	DL	**N**

The bold portions of the DDCF are updated by the attachment as each record is operated on:

N-Byte—Decremented by 1 after record orientation.

R-Byte—Incremented by 1 after record orientation if the last record specified was not operated on and if check status was not posted for the last record that was operated on. If head switching occurs, the R-byte is forced to hex 01 so that the first record read from the next track is record 1 (Note that record 0 is bypassed during head switching.)

H-Byte—The head number (second H-byte) is incremented by 1 at the index marker after record orientation if: (1) the last record specified by the instruction was not operated on by the drive (that is, if the N-byte does not hold FF) and (2) no check status except end of cylinder is posted.

End of cylinder status is only posted when the physical head number at the disk drive is incremented beyond hex 13. EOC status is *not* posted when the logical head number contained in the DDCF is incremented beyond hex 13:

- If any of the following check bits are posted, the record identifier portion of the DDCF contains the address of the last record being operated on:

	Byte	Bit
Format error	0	0
Missing address marker	0	2
Data check	0	4
No record found	0	5
Data overrun	0	7

- If end-of-cylinder status is posted, the R-byte and N-byte residuals are valid and can be reused when the data operation is restarted on a new cylinder.

The number of records processed can be derived from the residual value of the N-byte:

1. If N = hex FF, the specified number of records, N0 + 1, have been operated on (where N0 represents the value in the N-byte at the start of the operation).

2. If N = hex FF, the number of records operated on equals N0 - n (where N0 represents the value in the N-byte at the start of the operation, and n represents the residual value in the N-byte.

- If an equipment check is posted, the integrity of the DDCF cannot be guaranteed.

5445 Disk Drive Control Register (DDCR) Residuals

The disk drive control register is returned to its initialized value at the end of any operation in which it is used.

- If an equipment check is posted for the operation, the contents of the register are not guaranteed.

- If an end of cylinder (status byte 1, bit 5) is posted during a multiple record operation, the contents of the DDCR are equal to the initialized value plus 2.

5445 Disk Drive Data Field (DDDF) Residuals

DDDF residuals for any normal operation except read are identical with the initialized data. At the end of a write operation, the DDDF contains the data from the key and data fields of the specified record. If the instruction executed specified the reading of multiple records, the key and data fields of sequentially read records occupy contiguous positions of the DDDF without any indication of where one record ends and the next record starts.

5445 Disk Drive Data Register (DDDR) Residuals

At the end of scan and write count key data operations, the DDDR contains the initial value. At the end of data overrun operations, the DDDR contains the address of the last DDDF position acted upon. At the end of all other operations, the DDDR contains the address of the last DDDF position acted upon, plus 1.

Exercise care that the last DDDF position acted upon is not the byte immediately preceding the last byte location in storage. This results in an invalid address processor check when the DDDR is set to its final value.

5445 TIMINGS

5445 Disk Access Times

- Minimum—25 ms

- Average—60 ms

- Maximum—130 ms

For more exact access timings, see Figure 7-16.

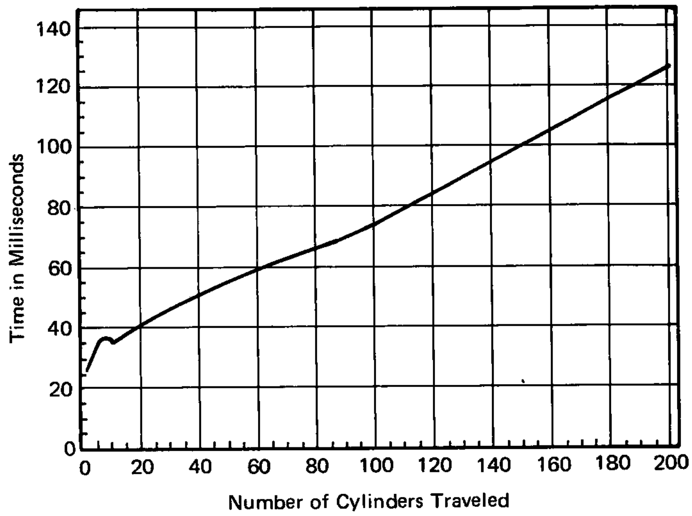


Figure 7-16. 5445 Disk Access Times

5445 Command Execution Times

Generally, command execution time represents that period of time during which the response to a test for I/O attachment busy is positive. The start I/O control commands specifying seek and recalibrate operations require additional seek busy time to complete mechanical motion and head switching. Head switching during multiple record operations also requires additional time. For rough timings, assume an average rotational delay time of 12.5 milliseconds, and a factor of 3.2 microseconds for each byte acted upon. See Figure 7-17 for the command execution timings formula that allows you to derive more exact command execution timings.

Command Type	Notes	Busy Times	
Control		**Seek Busy (Max/Min)**	**Attachment Busy (Max/Min)**
Seek			
Head Select	3	= 30/20 microseconds	30/20 microseconds
Head Motion		= 0.53(C-1) + 25 milliseconds where: $1 \leqslant C \leqslant 202$	34/26 microseconds
Recalibrate		130/25 microseconds	50/25 milliseconds
Read		Attachment busy	
HA and R0 Count	1, 4	= 269 microseconds	
KD	1, 2, 4	$= 3.2 \left\{ \sum_{\text{Record } 1}^{N+1} [\, DL + (KL + 45) \,] + 112\,N + 77 \right\}$ microseconds	
CKD	1, 4	= 3.2 [189 + 2DL + 2 (KL + 45)] microseconds	
Verify	1, 2, 4	Same as Read KD	
Write			
HA and R0	1	= 25 milliseconds	
C-K-D		Assume an average execution time of 12.5 milliseconds	
K-D	1, 2, 4	Same as Read KD	
Scan	1, 2, 4	Same as Read KD	

Notes:

1. Average rotational delay time of 12.5 milliseconds is not considered.
2. 522 microseconds must be added for each head switching action required for multiple record operations.
3. Seek busy for head motion is an approximate formula.
4. The term (KL + 45) must be set equal to zero when KL = 0.
   N = Number of records in excess of one to be operated on.

Figure 7-17. 5445 Command Execution Timings

5445 OPERATIONS

5445 Seek Operation

The seek control command selects one of 4,000 primary tracks or one of 60 alternate tracks on the disk drive specified by the DA- and N-code portions of the Q-byte. After a seek operation, a *cylinder* remains selected until a different cylinder is selected by a subsequent seek or recalibrate operation. A *track* remains selected until a different track is selected by a new seek or recalibrate operation or until automatic head switching occurs. (A track that is initially selected by a seek or recalibrate operation is changed by any subsequent multiple-record read, write, or scan command that causes automatic head switching to occur.)

The seek command does not verify that the correct track was selected. Invalid cylinder and head number checking is not performed.

A zero cylinder seek (that is, seek to the same cylinder) is provisionally accepted (stacked) while a seek or recalibrate command is being executed. The system executes the command at the end of the seek operation unless equipment check status (byte 0, bit 3) is posted.

Initial Conditions

DDCF—contains the 5-byte seek address format (FCCHH) used to specify the seek to cylinder and head number. The 5 remaining bytes in the DDCF are not used.

	Seek-to Address		Not Used for Seek Operation			
Flag	**Cylinder Number**	**Head Number**	Record Number	Key Length	Data Length	N

F—Not used.

CC—A 2-byte cylinder number field that specifies the cylinder number. Byte 1 should be hex 00, and the second byte must be the hexadecimal number of the cylinder. Cylinders are identified by decimal numbers 000 through 202, or hex 00 through CA. Cylinder numbers are not hardware checked.

HH—A 2-byte head number field that specifies the head number. The first byte should be set to 0. The second byte is set to the binary number of the seek to head. Decimal head numbers for byte 2 are 00 through 19, or hex 00 through 13. Head numbers are not hardware checked.

DDCR—Must contain the address of the leftmost (high-order) byte of the DDCF.

DDDF—Unchanged.

DDDR—Unchanged.

In Process Conditions

Test I/O—Selected device seek-busy response is positive until the seek operation is completed and the read head has settled enough to read data without errors.

Test I/O—Attachment busy is positive:

1. From the time the seek command is issued until the drive accepts the seek information

2. From provisional acceptance of a read, write, or scan command until the operation is completed

An overlapped seek operation can be initiated if the selected device is not seek busy or attachment busy.

Ending Conditions

DDCF—Remains unchanged.

DDCR—Contains the initialized address. The contents of the register are unpredictable if equipment check status is posted. See *5445 Disk Drive Control Register (DDCR) Residuals* in this section.

5445 Recalibrate Operation

The recalibrate control command starts a direct seek to cylinder 0 and head 0. Execution is the same as that for the seek command except for command execution times. Initial control and register fields need not be specified and therefore remain unchanged.

5445 Read Home Address and Record 0 Operation

Read home address (HA) and record 0 (R0) transfers all data from the 5-byte home address field (FCCHH) and all data from record 0 on the track under the active read head into main core storage. The 5445 locates the home address area, then reads the home address into the disk drive control field in main storage and reads record 0 into the disk drive data field in main storage. Record 0 key length and data length are obtained from the R0 count area on the actual disk.

Initial Conditions

DDCF—Destination field for the data in the first 5 bytes (FCCHH) of the home address area of the active track. (These bytes hold the flag data and the track address.)

DDCR—Must contain the address of the leftmost byte of the DDCF.

DDDF—Destination field for data from record zero (R0) of the active track. Field length = (key length + data length + 9).

DDDR—Must contain the address of the leftmost byte of the DDDF.

In Process Conditions

Busy to all commands except sense I/O or SIO interrupt until test I/O attachment busy is negative.

Ending Conditions

DDCF—Contains flag byte and track number from the home address area of the track.

DDCR—Contains the initial address. (If an equipment check is posted, the contents of the register are not guaranteed.)

DDDF—Contains data from record 0 count field.

DDDR—Contains the starting DDDR value, plus 9.

5445 Read Key Data Operation

The read key data operation transfers one or more disk records from the selected 5445 track into main storage. Reading begins at the record specified by the identifier field (CCHHR) in the disk drive control field in main storage. Record orientation is conditioned (that is, the correct record is assumed to have been found on the track) when the flag and identifier fields of a record on the disk track exactly match those fields in the disk drive control field (DDCF) located in main storage.

The key and data lengths need not be specified in main storage because these lengths are automatically read from the actual disk record by the attachment.

The attachment reads key and data fields into contiguous positions of the disk drive data field (DDDF) in main storage. The drive reads one more than the specified number of multiple fixed-format consecutive records (up to a maximum of 256 records) during this operation if the disk drive control field N-byte specifies a number greater than 0. As soon as each record is read, the attachment increments the DDCF record number byte (R) by 1, and decrements the DDCF N-byte (N) by 1.

When properly specified by the disk drive control field, record 0 (R0) on a track can be read. However, the drive bypasses R0 whenever R0 is encountered after head switching during multiple-record operation. During head switching operations, the attachment selects record 1 on the next sequential track as the next record to be read, thereby bypassing record 0.

Note: Head switching occurs at index time if record orientation was successful and multiple records are being read.

Initial Conditions

DDCF— Must contain the starting disk record address.

DDCR—Must contain the address of the leftmost byte of the DDCF located in main storage.

DDDF—Main storage area to receive the contiguous key and data fields from disk storage. Field length = (N + 1) (key length + data length).

DDDR—Must contain the address of the leftmost byte of the DDDF.

In Process Attachment Status

Attachment returns busy to all instructions except the SNS or SIO interrupt.

Ending Conditions

DDCF—Identifier portion contains the address of the last record read. The N-byte portion residual equals hex FF if all records were read.

DDDF—Contains contiguous key and data fields read from the disk.

DDDR—Contains the address of the last DDDF location operated on, plus 1; that is, disk drive data record 0 + (N + 1) (key length + data length), where the disk drive data record 0 is the initialized contents.

5445 Read Count Key Data Operation

This instruction recovers a single record under the following circumstances:

- The record being read has a defective count area.
- The key and data lengths of the record being read are unknown.

If record (Rn) is being read, the attachment starts the operation by orienting on record Rn-1 and then spaces over the following key and data fields of Rn-1. Reading begins at the next Rn count area. The drive transfers the first 9 bytes of the Rn count area into the DDCF in the CPU. Reading continues with the attachment using the key and data lengths extracted from the Rn count area, and transferring the contents of the key and data fields from disk record Rn into the DDDF.

Reading can begin at record R0 with the appropriate DDCF specification.

Initial Conditions

DDCF—Must specify the address of the disk record (Rn) to be recovered. The attachment orients on record Rn-1.

DDCR—Must contain the address of the leftmost byte of the DDCF.

DDDF—CPU field that receives the contents of contiguous key and data fields from the disk drive. Field length = (key length + data length).

DDDR—Must contain the address of the leftmost byte of the DDDF.

In Process Conditions

The attachment is busy to all commands except SNS and SIO interrupt. For command execution timings, see Figure 7-17.

Ending Conditions

DDCF—Identifier portion contains the address of the last record read.

DDCR—Contains the initialized leftmost byte address of the DDCF.

DDDF—Contains data read from count, key, and data fields on contiguous disk records.

DDDR—Contains the address of the last DDDF location operated on; that is, disk drive data record 0 + key length + data length + 9, where disk drive data record 0 is the initialized contents.

5445 Verify Key Data Operation

The verify key data operation performs a read back check of the key and data fields. This operation is the same as a normal read key data operation, except that data transfer does not take place. The attachment performs the read back check by comparing generated cyclic check and bit count appendage fields with the corresponding fields read from the selected disk. Key and data fields that are read are not compared.

To ensure that data was written accurately, issue a verify key data instruction immediately after any write command that modifies the key or data fields. Verification begins at the record specified by the identifier portion of the DDCF. The attachment reads the key length and data length from the count field of the record on the disk, so these lengths do not have to be supplied by the program. To verify multiple consecutive records, specify the number of records to be verified, plus 1, in the DDCF.

A maximum of 256 records can be verified without reissuing a new command.

Head switching can occur during command execution. However, during head switching operations, the drive starts examining records on the new disk at the index marker and searches until it encounters the record assigned the hexadecimal number 01 before it restarts the verification function. This means that record 0 is not verified.

Initial Conditions

DDCF—Must contain the address of the first record to be verified, and the number of records (N+1) to be verified.

DDCR—Must contain the address of the leftmost byte of the DDCF.

DDDF—Not used.

DDDR—Not used.

In Process Conditions

The attachment is busy to all commands except SNS and SIO interrupt. See Figure 7-17 for command timings.

Ending Conditions

DDCF—Identifier portion contains the address of the last record verified. The N-byte portion contains hex FF if all records were verified.

DDCR—Contains the initialized leftmost byte address of the DDCF.

DDDF—Remains unchanged.

DDDR—Remains unchanged.

5445 Write Home Address and Record 0 Operation

A write HA and R0 operation usually establishes track identity. Each track must be initialized with a write home address and record 0 operation before a data operation that involves record 0 can be performed. Thereafter, records written on the track must be numbered consecutively as the records are first written.

The write HA and R0 operation starts with the disk drive examining bit 1 of the DDCF (disk drive control field) flag byte. Then, when the drive senses the index marker, the drive writes the home address, record 0, and their associated gaps in the following sequence:

1. Gap 4. This gap contains 73 bytes if the flag byte bit 1 is 0, or 778 bytes if bit 1 is 1. (This data is generated by the drive.)

2. Data from the F, CC, and HH bytes of the DDCF.

3. Two cyclic check bytes, then a BCI (bit count indicator) and a BCA (bit count appendage) byte that are generated by the drive.

4. Gap 5, which is generated by the drive.

5. Record 0. The FCCHHR portion of the count field in the DDCF is used to format the count area of record 0. Then, the key and data fields for record 0 are written onto the disk track. As record 0 is written, the drive generates and writes gaps 1, 2, and three, as required.

   Note: R must be assigned the hexadecimal number 00 by the program to ensure correct disk operation. However, the drive does not check the program-assigned number during this operation.

6. After record 0 is written, the drive fills the remainder of the track with hex FF bytes.

Program Note

After the operation is complete, the program should issue a read home address and record 0 command. If a check status results during each of several successive rereads, the program should set the flag byte bit 1 to a 1, and reissue the write home address and record 0 command. The program should then assign an alternate track for the defective track, load the alternate track address into the count area of record 0 with a write count key data command (that indicates the primary track is defective), then write the entire record 0 onto the alternate track specifying the address of the defective track in the record 0 count area along with the indication that this is an alternate track. Flag byte bit 1 is not written on the disk record when the bit is being used for displacement control.

Initial Conditions

DDCF—Contains the FCCHHR KL DL DL N field specifications for HA and R0.

FCCHH—HA flag and home address.

FCCHHR—R0 count area flag and identifier.

KL DL DL—R0 key length and data length specifications.

N—Not used.

DDCR—Must contain the address of the leftmost byte of the DDCF.

DDDF—Contains contiguous R0 key and data fields.

DDDR—Must contain address of the leftmost byte of the DDDF.

In Process Attachment Status

The attachment is busy to all instructions except sense I/O.

Ending Conditions

DDCF—Contains the original contents with N unchanged.

DDCR—Contains the initialized address.

DDDF—The original contents are unchanged.

DDDR—Contains the address of the last DDDF position operated on plus 1.

5445 Write Count Key Data Operation

This is a single track initialization operation used to format single or multiple fixed format records (R0 through Rn). The disk drive starts formatting records at the record specified by the record identifier in the DDCF and formats n+1 records. The drive formats the count, key, and data areas as specified by the DDCF. The FCCHR of the count area is obtained from the DDCF. Key and data fields to be written are obtained from contiguous positions within the DDDF. Corresponding field length counts, KL and DL, are obtained from the DDCF. As the drive writes on the track, the attachment accumulates a KL + DL sum. A sum greater than 256 sets wrong length record (WLR) status and terminates the operation.

If record Rn is to be formatted, the attachment starts the operation by orienting on record Rn-1, then spaces over (but ignores) Rn-1. The drive then formats record Rn. After n+1 records are formatted, the remainder of the track is filled with hex FF bytes. For orientation on record Rn-1, corresponding CCHHR fields contained in the DDCF and the count area read from disk must compare. (The R-byte of the FCCHHR field contained in the DDCF is initially decremented by one for comparison with the corresponding ID field contained in Rn-1.) When record R0 is specified as the starting record, the drive orients on the last 2 bits of the home address flag byte. If the flag in main storage equals the flag on the disk, orientation occurs.

The attachment obtains track condition bits 6 and 7 from the flag byte in the DDCF. Bit 0 of the flag byte is always written as a zero in R0, and alternates from 0 to 1 in subsequent records.

5445 Write Count Key Data (Formatting) Operation

Multiple consecutive fixed-format records can be written on a single track by specifying an N-byte greater than zero. A write count key data command must be reissued for each track to be formatted. Track overrun status is posted if the read head encounters the index pointer before all the specified information is written on the track. The record number (R) in the DDCF is automatically incremented by one and the N-byte is decremented by one as each record is written. The source program is responsible for observing track capacity limitations. The program must verify initialization by issuing an independent read verify key data command in order to meet file performance specifications.

The key and data fields of one record are identical with those of all other records, because the DDDR contains its initial value at the end of formatting each record.

Initial Conditions

DDCF—Contains the initial control field bytes (FCCHHR KL DL DL N) used to specify the starting record address, key and data length counts and the number of records (n+1) to be written.

DDCR—Must contain the address of the leftmost byte of the DDCR.

DDDF—Contains the information for contiguous key and data fields of the record to be written.

DDDR—Must contain the leftmost byte address of the DDDF.

In Process Conditions

The attachment is busy to all instructions except sense I/O.

Ending Conditions

DDCF—Unchanged.

DDCR—Contains the initialized DDCF address.

DDDF—Contents remain unchanged.

DDDR—Contains the initialized DDDF address.

5445 Write Key Data Operation

The write key data operation transfers specified key and data fields from main storage to the selected disk drive and track. The attachment compares the flag and identifier field (FCCHHR) of the DDCF with the same flag, and identifier field of the count area read from the selected track. Comparison begins with the first count area read. A successful comparison is called record orientation. Following record orientation, the result of the count field comparison and field checking determines how the write operation proceeds.

If the DDCF counts are equal and field checking shows no errors, then writing begins in the key and data areas of the oriented record. A mismatch sets the no record found status and terminates the operation after field checking.

As the drive writes each record, it generates check field bytes and appends them to each key or data field, as required.

The drive writes multiple fixed format consecutive records if the DDCF N-byte is greater than 0. After initial orientation, the attachment decrements the N-byte by 1. When a multiple-record operation is specified, the attachment updates the DDCF by adding 1 to the record number (R-byte) and subtracting 1 from the N-byte as each record is operated on.

Writing can begin at record R0 if the DDCR R-byte in the DDCR specifies 0. However, the drive bypasses R0 if R0 passes the read head after head switching during a multiple-record operation.

Initial Conditions

DDCF—Contains the initial control field bytes (FCCHHR KL DL DL N). Specifies the starting record address, key and data length counts, and the number of records (n+1) to be written.

DDCR—Must contain the address of the leftmost byte of the DDCF.

DDDF—Contains contiguous key and data fields to be written into disk storage. Length = (N+1)(KL + DL)

DDDR—Must contain the address of the leftmost byte of the DDDF.

In Process Conditions

The attachment is busy to all instructions except SNS and SIO interrupt.

Ending Conditions

DDCF—Contains the address of the last record written or attempted to be written.

DDCR—Contains the initialized leftmost byte address of the DDCF.

DDDF—Contents remain unchanged.

DDDR—Contains the address of the last DDDF position operated on plus 1, or disk drive data record R0 + (n+1) (KL + DL)

5445 Scan Operations

A scan operation compares a record in main storage with a record stored on the disk drive. A scan under mask is implemented by inserting hex FF mask characters into positions of the storage argument that are to be masked out (that is, that are not to be compared).

Scan equal, scan high or equal, and scan low or equal operations are provided. A scan hit is a testable state within the test I/O instruction. A sense I/O instruction must be issued to determine if a scan equal condition is found during a scan equal operation or scan high or equal operation.

5445 Scan Key Data Equal

The scan key data equal operation compares the contents of the key and data fields read from the selected disk drive with a corresponding key and data comparison field argument in main storage. Comparison begins at the record specified by the identifier field (CCHHR) in the DDCF. Single or multiple-byte fields can be scanned under mask by inserting a mask character (hex FF) in byte positions of the main storage argument *not* to be compared. N+1 consecutive records can be scanned if the appropriate N-byte in the DDCF is specified. A maximum of 256 records can be scanned by a single instruction. After identifier orientation, the key-and-data-length-count fields specified determine how the scan proceeds: nonzero DDCF counts cause both key and data fields to be scanned. A mismatch between count fields sets no record found status and terminates the operation after field checking.

Scanning can begin at record R0 with the appropriate DDCF specification. However, R0 is bypassed if encountered after head switching during a multiple-record operation.

The scan operation proceeds until:

- A scan equal condition is found.
- N+1 records are scanned.
- An end-of-cylinder condition is detected.
- An equipment check or a data check is detected.

During the operation, the DDCF record number (R) is incremented by 1 and the N-byte decremented by 1 after each record is scanned. Multiple-record head switching occurs at index time provided record orientation is successful.

Initial Conditions

DDCF—Contains the starting record address.

DDCR—Must contain the address of the leftmost byte of the DDCR.

DDDF—Contains the comparison field argument. The DDDF is partitioned into key and data fields using the length counts specified in the DDCF.

DDDR—Must contain the address of the leftmost byte of the DDDF.

In Process Conditions

The attachment is busy to all instructions except SNS and SIO interrupt.

Ending Conditions

DDCF—Contains the address of the record in which a scan hit was found. Contains the address of the next record to be scanned if N+1 records are scanned and a scan hit is not found.

DDCR—Contains the address of the initialized leftmost byte of the DDCF.

DDDF—Remains unchanged.

DDDR—Contains the initialized address.

5445 Scan Key Data Low or Equal

This is a scan operation that is similar to scan equal. Field comparison results are based on low or equal conditions. A scan hit condition is set when the specified key and data fields read from the selected disk drive are lower than, or equal to, the masked argument in the DDDF. Test for a scan hit with a TIO instruction. A scan equal condition sets the scan equal status bit (byte 1, bit 6).

5445 Scan Key Data High or Equal

This is a scan operation similar to scan key data equal. Field comparison results are based on high or equal conditions. A scan hit condition is set when the specified key and data fields read from the selected disk drive are higher than, or equal to, the masked argument in the DDDF.

A scan hit can be tested via the TIO instruction. If a scan equal condition is found, the scan equal status bit (byte 1, bit 6) is set.

5445 Scan Read

The scan read command compares a data field read from the selected disk drive with a corresponding data comparison field (argument) in main storage.

Comparison starts at the record specified in the disk drive control field (DDCF), and continues until the first mark character (hex FF) is encountered in the argument. The data from the disk is then read into main storage.

If the first mask character is encountered on an even byte address and the argument is greater than 2 bytes, the next character that follows the first mark character remains unchanged. If the argument is not greater than 2 bytes, only the first mask character (hex FF) remains in main storage. If the first mark character is encountered on an odd byte address, only the mask character remains in main storage.

N+1 consecutive records (to a maximum of 256 records) can be scanned; the number of records to be scanned is specified by the N-byte in the disk drive control field (DDCF).

Program Notes

The scan read commands should not contain a key field because the key field is not read into main storage.

The scan read argument field in main storage does not require a hex FF (mask) character, but if one is used, the mask character must be located at least 2 bytes from the last character of the argument field. (The highest numbered storage location of the mask character equals data length minus 2).

A data overrun occurs if the mask character is not placed in a specified position in the argument field.

Scanning can start at record 0. However, R0 is bypassed if it is encountered after head switching during a multiple-record operation.

The scan read operation continues until:

- A scan hit occurs.
- N+1 records are scanned.
- End-of-cylinder is detected.
- An equipment check or data check is detected.

As the operation proceeds, the record number, R, contained in the DDCF is automatically increased by 1 and the N-byte is reduced by 1. Multiple-record head switching occurs at index time if record orientation is successful.

Initial Conditions

DDCF—Initialize the FCCHHR portion to the starting record address and the N-portion to the number of records in excess of 1 to be scanned. The DL need not be specified since it is obtained from disk count area.

DDCR—Initialize to the leftmost byte address of the DDCF.

DDDF—Comparison field argument. The data field using the length count read from the disk count area should be 2 bytes larger than the data field on the disk drive.

DDDR—Initialize to the leftmost byte address of the DDDF.

In Process Conditions

Attachment busy to all commands except sense I/O and SIO interrupt.

Ending Conditions

DDCF—Contains the record address in which a scan hit was found. Contains the address of the last record to be scanned if n+1 records were scanned and a scan hit was *not* found.

DDCR—Contains the initialized leftmost byte address of the DDCF.

DDDF—Contains the initial data including the first mask character, hex FF. If the first hex FF was found on an odd byte address, then the data from the disk is placed in the next sequential main storage address. If the first hex FF was located on an even byte address, the next character in main storage after the first hex FF also remains unchanged.

Examples:

- First hex FF was on odd-numbered address:

4000

DDDF contents at start of operation → | 01 | 02 | 03 | FF | FF | FF | FF |

Contents of data field on disk → | 01 | 02 | 03 | 04 | 05 | 06 | 07 | 08 |

DDDF contents at end of operation → | 01 | 02 | 03 | FF | 04 | 05 | 06 | 07 |

4000

- First hex FF was on even-numbered address:

4001

DDDF contents at start of operation → | 01 | 02 | 03 | FF | FF | FF | FF | FF |

Contents of data field on disk → | 01 | 02 | 03 | 04 | 05 | 06 | 07 |

DDDF contents at end of operation → | 01 | 02 | 03 | FF | FF | 04 | 05 | 06 | 07 | 08 |

4001

DDDR—Contains the address initialized.

Ending Status

End-of-cylinder status is not posted if the operation is ended prior to EOC detection.

5445 START I/O (SIO)

Op Code (hex)	Q-Byte (binary)	R-Byte (binary)
Byte 1	Byte 2	Byte 3
F3	110x x xxx	xxxx xxxx

Q-byte: DA = 110x, M = x, N = xxx; R-byte: Control Code

N-Code	Bits 0123 4567	Function Specified
000	0000 0000	Seek
	0000 0001	Recalibrate
001	0000 0000	Read key data
	0000 0001	Read home address and record 0
	0000 0010	Read count key data special
	0000 0011	Read verify key data
	0000 0100	Read count key data diagnostic (CE diagnostic)
	0000 0111	Read buffer diagnostic (CE diagnostic)
010	0000 0000	Write key data
	0000 0001	Write home address and record 0
	0000 0010	Write count key data
011[1]	0000 0000	Scan key data equal
	0000 0001	Scan key data low or equal
	0000 0010	Scan key data high or equal
	0000 1000	Scan read equal[2]
	0000 1001	Scan read low or equal[2]
	0000 1010	Scan read high or equal[2]
100[1]	1xxx xx00	Enable interrupt for all 5445s[2]
	x1xx xxx0	Reset seek 1 interrupt[2]
	xx1x xxx0	Reset seek 2 interrupt[2]
	xxx1 xxx0	Reset seek 3 interrupt[2]
	xxxx 1xx0	Reset seek 4 interrupt[2]
	xxxx x1x0	Reset op end interrupt for all 5445s[2]
	0xxx xx10	Disable interrupt for all 5445s

Any N-code not shown is invalid and causes:
- Program check if interrupt level 7 is enabled on Model 15
- Processor check if interrupt level 7 is not enabled on Model 15
- Processor check on Model 10

x = Can be 1 for multiple control instructions.

Any control code not shown may result in the attachment hanging up in a busy state.

DA = 1100 and M = 0 specifies 5445 drive 1 as the addressed unit.
DA = 1100 and M = 1 specifies 5445 drive 2 as the addressed unit.
DA = 1101 and M = 0 specifies 5445 drive 3 as the addressed unit.
DA = 1101 and M = 1 specifies 5445 drive 4 as the addressed unit.
(Note that Q-bits 0, 1, 2, = 110 specifies the 5445, while Q-bits 3 and 4 specify the drive)

F3 specifies a start I/O operation. F as the first hex character in the op code identifies a command-type instruction (that is, an instruction without operand addressing).

[1] Q-byte bits 3 and 4 (drive specification bits) are ignored; attachment circuits are addressed.
[2] This is an invalid N-code on Model 10.

Operation, General

The drive specified by the DA- and M-codes performs the function specified by the N-code and control code. *Exception:* When the N-code = 100, the SIO commands specify interrupt control. In this case, the command addresses all installed drives although seek interrupts still apply to individual, specified drives.

Program Notes

- Issuing any start I/O except interrupt control, read diagnostic sense, read extended sense, or read data module control to a busy attachment causes the program to loop on the instruction until the attachment becomes not-busy. If the instruction addresses a drive that is not installed, a program check or processor check occurs with an invalid Q-byte indicated.

- The attachment provisionally accepts a single start I/O specifying read, write, or scan for later execution whenever the addressed drive is executing a seek. If an error occurs during the seek, the attachment aborts the provisionally accepted SIO. At the end of the seek operation, the attachment then sets no-op status bit, the unit check bit, and either a seek check bit or attachment check bit (as appropriate), and requests an op-end interrupt.

- A seek instruction on one drive can be overlapped with seek instructions on all other drives. A read, write, or scan on one drive can be overlapped with a seek instruction on any other drive if the seek instruction is issued first. Overlapping does not occur if the seek is issued during a read, write, or scan operation on any drive.

- The start I/O instruction uses the contents of the disk drive (data) address register (DDDR) as the initial main storage address of all disk record data fields. It uses the contents of the disk drive control (address) register as the address of the disk drive control field (DDCR) in main storage.

- The attachment always accepts an SIO interrupt control instruction, regardless of the status of the file or control unit. Issuing this SIO *does not* reset the attachment status.

Op-End Interrupts (Model 15)

The attachment presents an op-end interrupt request to the Model 15 processing unit at the end of the processing unit instruction during which one of the following conditions occurred on the selected drive:

- The drive completed a data transfer operation (either read, write, or scan).

- The drive finished a seek operation.

- A read, write, or scan SIO was aborted because of an equipment check.

- An attachment check is pending.

Note: The attachment does not post an op-end interrupt at the end of either a read extended functional sense operation or a data module attention control reset operation.

5445 LOAD I/O (LIO)

Op Code (hex)	Q-Byte (binary)	Operand Address	
Byte 1	**Byte 2**	**Byte 3**	**Byte 4**
31	110x x xxx	Operand 1 address	
71	110x x xxx	Op 1 disp from XR1	
B1	110x x xxx	Op 1 disp from XR2	

DA M N

N-Code	To Be Loaded
100	Disk drive data register (DDDR)
101	CE diagnostic LIO 1
110	Disk drive control register (DDCR)
111	CE diagnostic LIO 2

Any N-code not shown is invalid and causes:
- Program check if interrupt level 7 is enabled on Model 15
- Processor check if interrupt level 7 is not enabled on Model 15
- Processor check on Model 10

DA = 1100 and M = 0 specifies drive 1.
DA = 1100 and M = 1 specifies drive 2.
DA = 1101 and M = 0 specifies drive 3.
DA = 1101 and M = 1 specifies drive 4.

Hex 31, 71, or B1 specifies a load I/O operation. The first hex character in the op code specifies the type of operand addressing for the instruction.

Operation

The processing unit loads the 2 bytes of data contained in the operand into the register specified by the N-code. The operand is addressed by its low-order (higher numbered) storage position.

Program Notes

- An LIO with an N-code of 100 or 110 issued to a busy attachment causes the program to loop on the LIO until the attachment is no longer busy.
- LIO does not set any disk status conditions.
- LIO is executed if the addressed drive is executing a seek or recalibrate operation and a read, write, or scan *was not* accepted or provisionally accepted.
- An LIO with an N-code of 100 or 110 is always executed unless the no-op bit is on.
- Exercise care when loading the DDDR so that the last position of the DDDF is not coincident with the byte immediately preceding the last byte location in storage. This results in an invalid address processor check when the DDDR is set to its final value.

5445 TEST I/O AND BRANCH (TIO)

Op Code (hex)	Q-Byte (binary)	Operand Address	
Byte 1	Byte 2	Byte 3	Byte 4
C1	110x x xxx	Operand 1 address	
D1	110x x xxx	Op 1 disp from XR1	
E1	110x x xxx	Op 1 disp from XR2	

DA M N

N-Code	Condition Tested
000	Not ready/unit check
001	Seek busy
010	Attachment busy
011	Scan hit
100	*Model 10:* Invalid N-code *Model 15:* Interrupt pending

Any N-code not shown is invalid and causes:
- Program check if interrupt level 7 is enabled on Model 15
- Processor check if interrupt level 7 is not enabled on Model 15
- Processor check on Model 10

DA = 1100 and M = 0 specifies drive 1 as the tested unit.
DA = 1100 and M = 1 specifies drive 2 as the tested unit.
DA = 1101 and M = 0 specifies drive 3 as the tested unit.
DA = 1101 and M = 1 specifies drive 4 as the tested unit.

C1, D1, or E1 specifies a test I/O and branch operation. The first hex character in the op code specifies the type of operand addressing for the instruction.

Operation

The processing unit tests the drive specified by the DA- and M-codes for the condition specified by the N-code. If the condition exists, the program branches to the location specified by the operand address. If the condition does not exist, the program advances to the next sequential instruction.

Resulting Condition Register Setting

This instruction does not affect the condition register.

IAR and ARR Contents after Instruction Execution (Model 15)

If the branch occurred, the IAR contains the branch-to address (from the operand address of the instruction) and the ARR contains the address of the next sequential instruction.

If the branch did not occur, the IAR contains the address of the next sequential instruction and the ARR contains the branch-to address from the operand address of the instruction.

The information stored in the ARR remains there until the next decimal, insert-and-test-characters, branch, or test-I/O instruction is executed.

Program Notes

- Unit check indicates that the addressed disk drive has either a disk drive check status or a common check status outstanding. A common check relates to those sections of the attachment that are shared by all the drives. The usual checks are:

  Command reject
  Invalid track format
  Instruction required
  Track condition check
  Equipment check
  Data check
  No record found
  Write inhibited
  Data overrun
  Command overrun
  Environmental data present
  End of cylinder
  Seek check

  A seek check for the drive not addressed is not indicated. The drive that has the check condition can be determined from the attachment sense bytes.

- Seek busy indicates that the addressed disk drive is performing a seek or recalibrate operation.

- Attachment busy indicates that either the addressed disk drive or attachment:
  - Is executing a read, write, or scan instruction
  - Is in the starting phase of the seek operation that requires additional CPU cycle steal requests
  - Has provisionally accepted a read, write, or scan instruction for subsequent execution, or
  - Is currently involved in an IMPL operation.

- Scan hit indicates that a previously issued scan command caused data transfer. Scan hit is an indication that is common to all drives; that is, a scan hit on any drive is always indicated to the program, no matter which drive was addressed in the TIO instruction. For example, if a scan command to drive 1 resulted in a scan hit, and drive 2 is addressed by a TIO instruction that specifies testing for a scan hit, a branch occurs.

5445 ADVANCE PROGRAM LEVEL (APL)

Op Code (hex)	Q-Byte (binary)	R-Byte (binary)
Byte 1	Byte 2	Byte 3
F1	110x x xxx	0000 0000

DA M N

R-byte is not used in an APL instruction.

N-Code	Condition Tested
000	Not ready/unit check
001	Seek busy
010	Attachment busy
011	Scan hit
100	*Model 10:* Invalid N-code; causes processor check *Model 15:* Interrupt pending

Any N-code not shown is invalid and causes:
- Program check if interrupt level 7 is enabled on Model 15
- Processor check if interrupt level 7 is not enabled on Model 15
- Processor check on Model 10

DA = 1100 and M = 0 specifies drive 1 as the tested unit.
DA = 1100 and M = 1 specifies drive 2 as the tested unit.
DA = 1101 and M = 0 specifies drive 3 as the tested unit.
DA = 1101 and M = 1 specifies drive 4 as the tested unit.

F1 specifies an APL operation. F as the first hex character in the op code identifies a command type instruction (that is, an instruction without operand addressing).

Operation

This instruction tests for the conditions specified in the Q-byte.

- Condition present:
  - Systems with Dual Program Feature installed and enabled, activate the inactive program level.
  - Systems without Dual Program Feature installed or with Dual Program Feature installed but not enabled, loop on the advance program level instruction until the condition no longer exists.

- Condition not present: Systems with or without Dual Program Feature access the next sequential instruction in the active program level.

Program Note

For additional information concerning the advance program level instruction, see Chapter 2.

5445 SENSE I/O (SNS)

Op Code (hex)	Q-Byte (binary)	Operand Address	
Byte 1	Byte 2	Byte 3	Byte 4
30	110x x xxx	Operand 1 address	
70	110x x xxx	Op 1 disp from XR1	
B0	110x x xxx	Op 1 disp from XR2	

DA M N

N-Code	Sensed Unit
000	Status bytes 0 and 1
001	Status bytes 2 and 3 (CE diagnostic)
010	Status bytes 4 and 5 (CE diagnostic)
011	Status bytes 6 and 7 (CE diagnostic)
100	Disk data address register (DDDR)
101	Status bytes 8 and 9 (CE diagnostic)
110	Disk control field address register (DDCR)

Any N-code not shown is invalid and causes:
- Program check if interrupt level 7 is enabled on Model 15
- Processor check if interrupt level 7 is not enabled on Model 15
- Processor check on Model 10

DA = 1100 and M = 0 specifies 5445 drive 1 as the sensed unit.
DA = 1100 and M = 1 specifies 5445 drive 2 as the sensed unit.
DA = 1101 and M = 0 specifies 5445 drive 3 as the sensed unit.
DA = 1101 and M = 1 specifies 5445 drive 4 as the sensed unit.

Q-byte bits 0, 1, and 2 = 110 specifies the 5445 attachment as the unit being sensed. Bits 3 and 4 can be any value.

30, 70, or B0 specifies a sense I/O operation. The first hex character in the op code specifies the type of operand addressing for the instruction.

Operation

The attachment transfers 2 bytes of data to the main storage field specified by the operand address. The first byte transferred (the odd-numbered status byte) enters high-numbered storage position in the operand; the other byte enters the low-numbered position of the operand, which is specified by the operand address.

The drive accepts a sense I/O instruction at any time, even though another operation may be in progress when the instruction is issued. See Figure 7-18 for an explanation of the status bits.

Program Notes

- The sense instruction resets the no-op status bit at the end of the sense operation.
- The end-of-cylinder status bit is not valid unless the SNS instruction was issued while the attachment was *not busy* (5445 only).

Byte	Bit	Name	Indicates	Reset By
0	0	Format error	Program attempted to format beyond disk capacity (key length plus data length was more than 256 bytes) *or* beyond the track capacity.	Next SNS or SIO instruction, or system reset
0	1	Intervention required	Drive is powered down, is in a start-up transition, or is switched offline.	Correcting the condition that set the bit; usually requires manual intervention
0	2	Missing address mark	Disk needs to be reinitialized.	Next SNS or SIO instruction or system reset
0	3	Equipment check	The selected drive detected an unsafe condition; *or* the control unit detected a parity, serial, cyclic, or BCA check; *or* the selected drive went not-ready (setting the no-op status bit) while the attachment was still busy.	Next SNS or SIO instruction or system reset
0	4	Data check	The attachment discovered an error in a home address, count, key, or data field.	Next SNS or SIO instruction or system reset
0	5	No record found	One of the following: – The record specified by the ID field could not be found. – During a multiple-record operation, one of the records following the first record could not be found. – The missing address mark bit was set (see *status byte 0, bit 2*).	Next SNS or SIO instruction or system reset
0	6	No-op	An SIO that specified a function other than recalibrate was issued to a disk drive with an outstanding seek function.	Next SNS or SIO instruction or system reset
0	7	Data overrun	The I/O channel cycle steal request was not granted in time to maintain the required data transfer rate. *This should never occur.*	Next SNS or SIO instruction or system reset
1	0	Disk drive error	The selected drive detected an unsafe or seek incomplete condition.	Correcting the condition that set the bit; usually requires manual intervention
1	1	Unsafe	A condition exists that prevents the drive from ensuring data integrity.	Correcting the condition that set the bit; usually requires manual intervention
1	2	Seek 1 complete	Interrupt was enabled and a programmed seek was completed on 5445 drive 1.	System reset, check reset, or next SIO that resets seek 1 interrupt
1	3	Seek 2 complete	Interrupt was enabled and a programmed seek was completed on 5445 drive 2.	System reset, check reset, or next SIO that resets seek 2 interrupt

Figure 7-18 (Part 1 of 2). 5445 Disk Drive Status Bytes

Byte	Bit	Name	Indicates	Reset By
1	4	Op-end	A read, write, or scan operation was terminated while op-end interrupt was enabled (Model 15A only).	System reset, check reset, or next SIO that enables or resets op-end interrupt
1	5	End of cylinder	The physical head number at the selected drive was incremented beyond head 19 during a multiple-record operation. That is, the operation specified more records than remained on the cylinder.	Next SIO instruction accepted by 5445
1	6	Scan equal	A scan equal condition was found during execution of an SIO scan operation.	Next SIO command accepted, system reset, or check reset
1	7	Disk drive identifier	5445 drive 1 or drive 3 was selected if this bit is not on; if this bit is on, 5445 drive 2 or drive 4 was selected. This bit is set only if interrupts are enabled.	System reset (next SIO accepted may change state of this bit)
2	0	CE diagnostic		
2	1	CE diagnostic		
2	2	Seek 3 complete	Interrupt was enabled and a programmed seek was completed on 5445 drive 3.	System reset, check reset, or next SIO that resets seek 3 interrupt
2	3	Read parity error	The attachment detected a parity error in data being transmitted to the processing unit from the active drive.	Next SIO accepted by 5445
2	4	Disk busy	The selected disk drive access mechanism is in motion or seek is busy for the selected disk.	Drive mechanism settling (motion stopping) or seek going not busy
2	5	CE diagnostic		
2	6	Seek 4 complete	Interrupt was enabled and a programmed seek was completed on 5445 drive 4.	System reset, check reset, or next SIO that resets seek 4 interrupt
2	7	Interrupt not pending	No interrupt is pending (op-end or seek). (Model 15A only.)	System reset, check reset, or next SIO that resets interrupt pending status

Figure 7-18 (Part 2 of 2). 5445 Disk Drive Status Bytes

IBM 3340/3344 Direct Access Storage Facility

3340 on System/3 Model 15B, Model 15C and 3344 on Model 15D

The IBM 3340/3344 Direct Access Storage Facility provides a maximum of 515 megabytes of direct access storage. Each Model B, C, and D must be equipped with one 3340 Model A2 and can also be equipped with either a 3340 Model B1 or B2. The Model D may be equipped with one 3344 Model B2 instead of a 3340 Model B1 or B2. A Model B, C, or D system excludes attachment of both the IBM 5444 and the IBM 5445.

Two, three, or four drives can be attached to a single system in the following configurations of 3340/3344 models:

Configuration of Models	Total Number of Drives	Total Capacity in Data Bytes
One 3340 Model A2 only	2	102,924,288
One 3340 Model A2 and one 3340 Model B1	3	154,386,432
One 3340 Model A2 and one 3340 Model B2	4	205,848,576
One 3340 Model A2 and one 3344 Model B2	4	515,801,088

Each 3340 drive contains the mechanical and electrical components needed to house, load, filter, and drive an IBM 3348 Data Module Model 70, which is a removable and replaceable disk pack. Each 3344 drive contains the mechanical and electrical components needed to house, filter, and drive a fixed media storage spindle. The IBM 3340 Model A2 additionally provides logic and power for all the 3340/3344 drives.

Attachment logic, standard in each IBM 5415 Model B, C, and D Processing Unit, is the interface between the processing unit and the 3340 control logic (controller).

3340 on System/3 Model 12

This system uses one 3340 Model C2 only. The Model C2 contains two drives and is identical in capacity to the Model A2 above; however, the control unit is internal to the 5412 processing unit.

IBM 3344 Direct Access Storage on System/3 Model 15D

The 3344 Direct Access Storage Model B2 is a two-drive unit that attaches to the 3340 Direct Access Storage Facility Model A2 on a 5415 Processing Unit Model D. The 3344 storage medium is permanently mounted and sealed within the drive as an integral component. Each spindle contains eight disks, 31 heads, with two heads per disk surface, except for the lower surface which has only one head that is used as a tracking head. Fifteen disk surfaces are available for reading and writing data and one surface is used for tracking.

The two drives of a 3344 provide approximately 365.7 million bytes of user data storage and 47.2 million bytes for programs, backup, and reserved areas. The disk assembly rotates at 2,964 rpm. Figure 7-19 shows one of the drives, reflecting the disk arrangement and head identification.

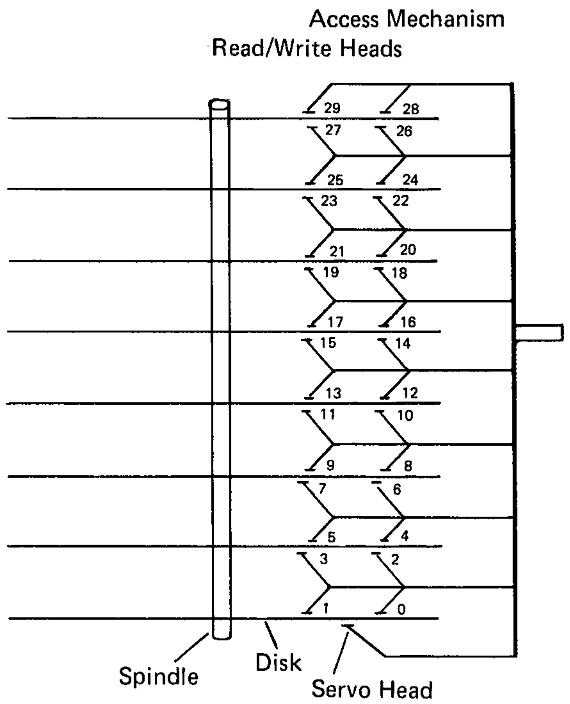


Figure 7-19. 3344 Disk/Head Layout

IBM 3348 DATA MODULE (DM) MODEL 70

The IBM 3348 Data Module (Figure 7-20) is a compact disk assembly; it contains four disks mounted on a spindle, and has a data access mechanism (Figure 7-21).

The data access mechanism consists of 12 read/write heads mounted on three access arms; each arm is firmly fastened to a movable carriage. Figure 7-22 shows that each of the arms is halfway between two disks, with two heads for the disk above the arm and two heads for the disk below the arm. The entire access mechanism can be positioned at 350 discrete positions, allowing the heads to access 350 outer tracks and 350 inner tracks on each of six disk recording surfaces.

Figure 7-20. IBM 3348 Model 70 Data Module

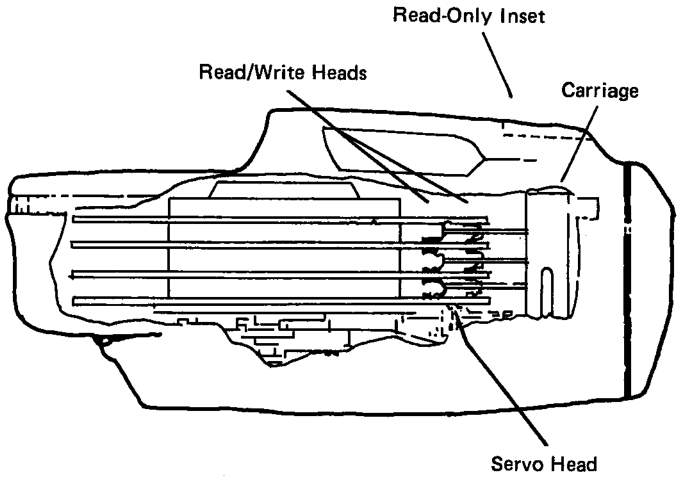


Figure 7-21. 3348 Model 70 Data Module Schematic

After you install the data module in the 3340, the drive engages the spindle and access carriage. (During normal data processing operations the drive and the installed data module operate as a single I/O device; therefore, this manual usually discusses operation of the data module as if it were part of the addressed 3340 drive.)

Note: Although other models of the 3348 fit the 3340, a 3340 attached to the System/3 uses only the 3348 Model 70 for customer data storage. (IBM customer engineers, however, use a special 12-megabyte data module for diagnostic operations.)

3348 Data Module Organization

The data module used by the 3340 is organized physically as shown in Figure 7-22. As you examine the figure, note that each disk surface has two index points and 700 tracks. The attachment maps the data module into a logical organization that supports addressing similar to IBM 5445 disk addressing on the System/3.

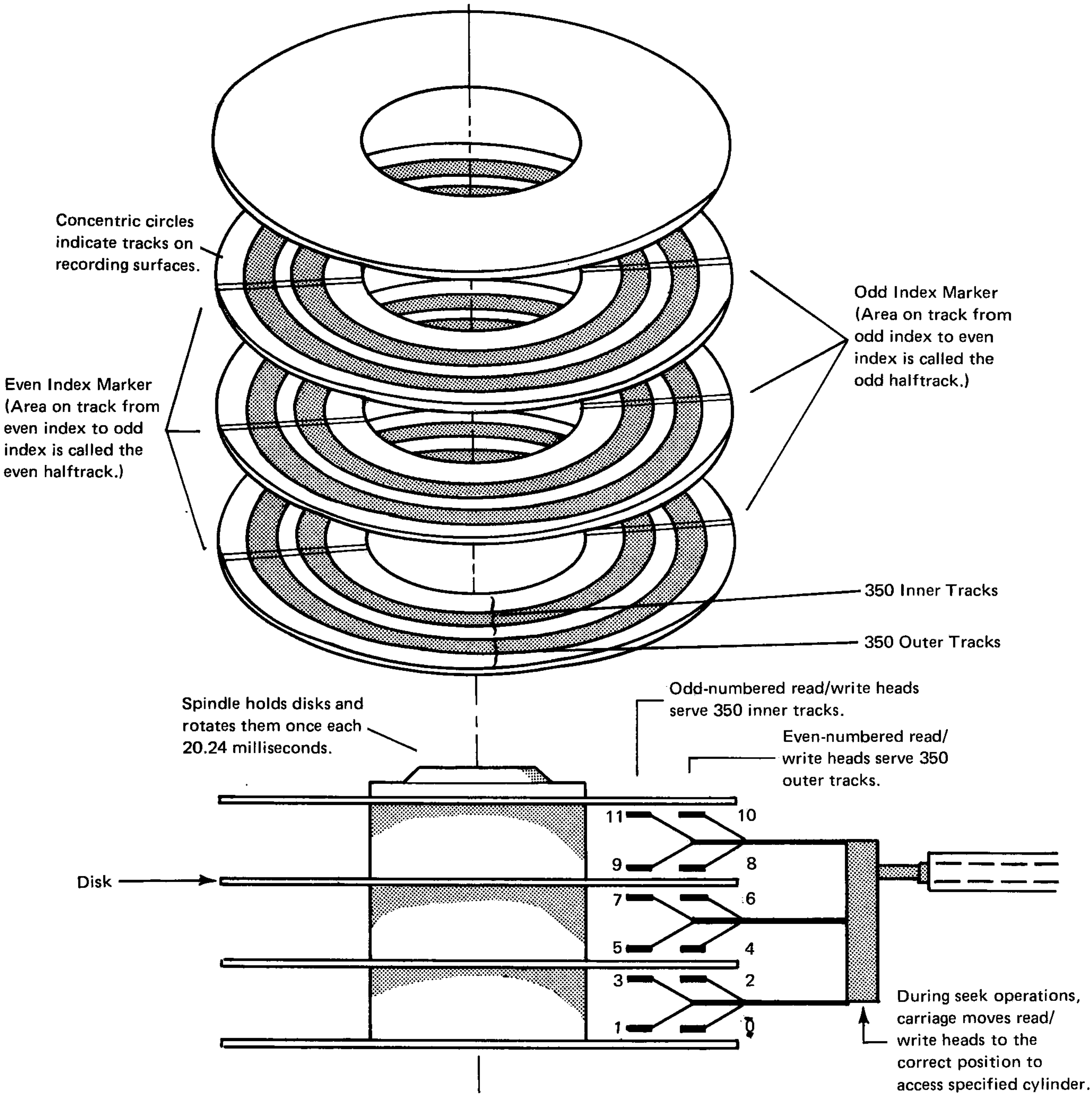


Figure 7-22. 3348 Data Surface Physical Track Layout

3340/3344 Physical Tracks

The area on the surface of the disk that passes each read/write head as the disk revolves at a single access position is called a *track.* Each surface has two tracks for each access position, because there are two read/write heads per disk surface.

Note that each track has an odd and an even index marker. This lets the 3340/3344 divide each track into an even and an odd halftrack. Each halftrack has a home address and a record 0 that always occupy the first two addressable areas on the associated halftrack and that are identified by the type of halftrack on which they are written. That is, after a track has been formatted it always has an even home address, then an even record 0 as the first two formatted areas following the even index marker (the even halftrack), and an odd home address, then an odd record 0 on the halftrack following the odd index marker (the odd halftrack). The remaining area on the entire track holds records that are sequentially numbered from the even record 0.

Attachment logic requires that both halftracks be reassigned to the same alternate track if either halftrack is reassigned an alternate, and that both of these reassigned primary half-tracks be flagged as defective.

3340/3344 Physical Cylinders

A physical cylinder can be defined as all the tracks that can be read by the drive from a single access position. Therefore, a physical cylinder consists of two tracks read from each of six disk surfaces. Because the 3340 has 350 access positions, it has 350 physical cylinders. The 3344 has 560 access positions and 560 physical cylinders.

3340/3344 Logical Cylinders

Attachment logic divides the 3348 data module and 3344 data storage into 210 logical cylinders; these are numbered from hex 00 through D1. To make the addressing similar to 5445 addressing on the System/3, attachment logic assigns 20 System/3 logical tracks to each logical cylinder.

3340/3344 Logical Tracks

The logical tracks are identical to the physical tracks, but are identified by logical head numbers and logical cylinder numbers. Each logical track can contain a maximum of 102 variable-length records (including the record 0 from each halftrack). When IBM System/3 programming support is used, each track can hold a maximum of 48 fixed length 256-byte records, plus two standard record 0's. (A standard record 0 has a key length of 0 and a data length of 8.)

3340 Logical Heads on 3348-70 Data Module

The 3348-70 data module has only 12 physical heads, but attachment logic must address 20 heads to make addressing compatible with IBM 5445 addressing. To achieve this, attachment logic provides an automatic overflow from one cylinder to the next, and uses 1-2/3 physical cylinders (with their associated physical heads) for each logical cylinder. Attachment logic assigns logical head numbers hex 00 through 13 to the 20 heads serving each logical cylinder.

ADDRESSING 3340/3344 TRACKS, CYLINDERS, AND RECORDS ON SYSTEM/3

The 3340/3344 disk addressing scheme is similar to that used for the IBM 5445 Disk Storage. Tracks are identified by cylinder number and head number, and records are numbered sequentially from the even index home address. Each home address and record 0 on a track is further identified to the attachment by the instruction issued (see *Read HA and R0 and Write HA and R0* in this section).

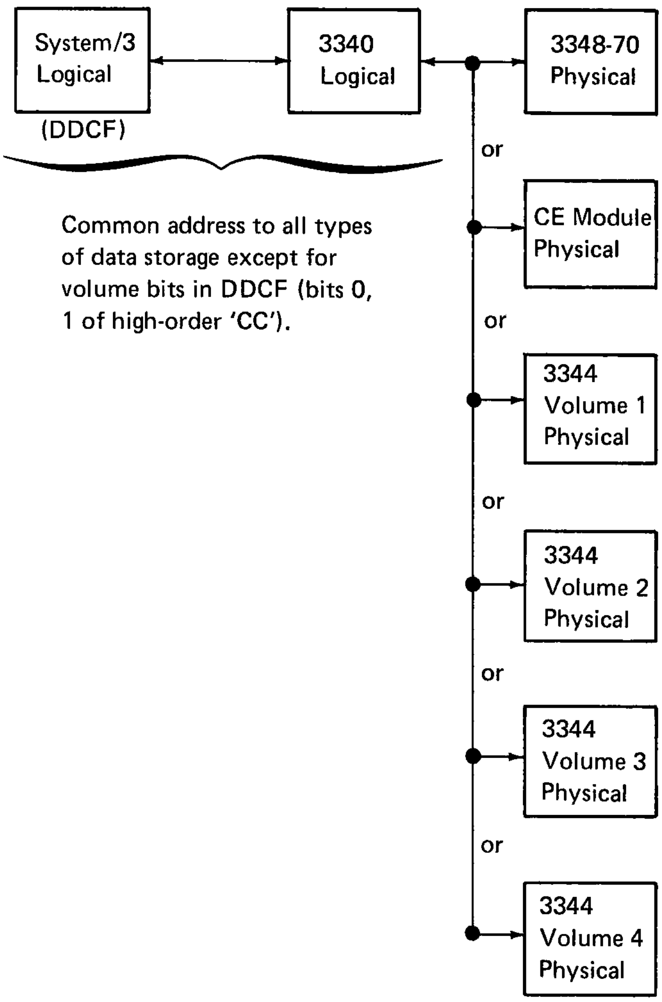
Attachment logic provides the necessary translation (from logical addresses to physical addresses) to access the correct cylinders and find the correct record.

Because logical addresses are used by the programmer, this manual refers to logical addresses when physical addresses are not specified.

3340/3344 Address Conversion

To allow programs written for the 5445 Disk Storage Drive to operate the 3340 Direct Access Storage Facility, the System/3 addresses the 3340/3344 as if it were a 5445. To do this, the 3340/3344 disk storage attachment microprocessor converts 5445 addresses to 3340/3344 addresses. The conversion is transparent to the user, but there are times when the CE must translate System/3 addresses to 3344 logical and 3340/3344 physical addresses and also convert 3340 logical and 3340/3344 physical addresses to System/3 addresses.[1]

Microcode Address Conversions



To understand the conversions you must first understand certain addressing terminology, described in the following paragraphs.

A *3348-70 physical track* is the area on the disk surface immediately under or above one physical read/write head during one revolution of the disk surface. There are two physical tracks, one for each physical head, on each of six disk surfaces. This makes 12 physical tracks for each of the 350 access positions. (See Figure 7-22.)

A *3344 data storage physical track* is the area on the disk surface immediately under or above one physical read/write head during one revolution of the disk surface. There are two physical tracks, one for each physical head, on each of 15 disk surfaces. This makes 30 physical tracks for each of the 560 access positions. (See Figure 7-19.)

A *3348-70 physical cylinder* is all physical tracks that can be read from a single access position. Since each data module has 350 access positions, there are 350 physical cylinders.

A *3344 data storage physical cylinder* is all physical tracks that can be read from a single access position. Since each 3344 spindle has 560 access positions, there are 560 physical cylinders.

A *3340 or 3344 logical track* is one-half of a physical track. The even-numbered logical tracks start at the even index point. The odd-numbered logical tracks start at the odd index point.

A *3340 logical cylinder* is one-half of a 3348-70 physical cylinder and one-fifth of a 3344 physical cylinder. That is, one access position on the 3348-70 data module contains two 3340 logical cylinders, while one access position on the 3344 contains five 3340 logical cylinders.[1] There are 700 3340 logical cylinders per 3344 data storage logical volume.

[1]Refer to IBM *3340/3344 Disk Storage Attachment for System/3 Models 12 and 15 Theory-Maintenance Diagrams*, SY31-0406, for a complete set of address conversion tables.

3340/3344 Track Capacity

Track Capacity in Standard Data Format

Each record must lie entirely in either the odd halftrack or the even halftrack. Therefore, the track capacity must be calculated by considering each halftrack capacity separately.

When using standard track format, the number of equal-length records that can be written on a halftrack depends upon the record length (Figure 7-23). For record lengths not shown, you can use the following equation. The equation takes the home address, standard R0, and skip defect areas into consideration:

$$\text{Number of equal-length records per halftrack} = \frac{8535}{C + KL + DL}$$

where:

8535 = halftrack capacity
C + KL + DL = bytes per record
C (overhead per record) = 167 if KL is 0
= 242 if KL is not 0
KL = key length
DL = data length
Overhead = bytes used for record formatting (in gaps)

Track Capacity in Compressed Data Format

Record 0 is never written in compressed data format. For all other records on the track, the key length must be 0 and the data length must be decimal 256 when writing in compressed data format. Therefore, (assuming that even R0 and odd R0 are written with a key length of 0 and a data length of 8) each track holds a maximum of 48 compressed format records for a total of 12,288 bytes of data.

IBM System/3 programming support always writes normal data tracks in compressed format with each R0 having the standard key length of 0 and data length of 8.

Record Length (Bytes)	Track Capacity	
	In Records	In Data Bytes
256	40	10,240
239	42	10,038
220	44	9,680
204	46	9,384
188	48	9,024
174	50	8,700
161	52	8,372
149	54	8,046
137	56	7,672
127	58	7,366
117	60	7,020
108	62	6,696
99	64	6,336
91	66	6,006
84	68	5,712
76	70	5,320
70	72	5,040
63	74	4,662
57	76	4,332
51	78	3,978
46	80	3,680
41	82	3,362
36	84	3,024
31	86	2,666
26	88	2,288
22	90	1,980
18	92	1,656
14	94	1,316
10	96	960
7	98	686
3	100	300

Figure 7-23. Track Capacity When Using Equal Length Records Written in Standard Format with KL = 0

3340 DATA SECURITY AND PRIVACY

The 3348 has a read-only function. This function, in conjunction with methods such as seek verification, offers a way to limit access to data areas of the data module.

3348 Data Module Read-Only Function

Each data module is equipped with a two-position switch in its handle that the operator can set to its appropriate setting before inserting the data module into the drive.

The READ-ONLY setting of the switch prevents the program from writing onto any disk in the module attachment logic rejects any write command addressed to that drive and (on Model 12 only) causes the CPU I/O ATTENTION light to turn on. The attachment returns command reject (sense byte 0, bit 0) and write protect (sense byte 1, bit 6) to a subsequent read diagnostic sense command issued to the same drive (drive must become not-ready on the Model 12).

When the drive becomes ready, a light on the drive indicates whether or not the read-only option has been selected for the installed data module.

3344 Write Protect Function

The write protect function is provided on the 3344 by a R/W or READ switch, located on the operator panel. When this switch is in the READ position, no write operation can be done; if set to R/W, all normal operations are possible. If the switch is changed during an operation, the condition does not change until the operation is completed. When the operation is complete, the movement of the R/W or READ switch is sensed by the attachment, and a data module attention condition is presented to the system. A data module attention condition is also presented to the system if the R/W or READ switch is set to READ while performing an initial microprogram load.

3340/3344 Seek Verification

The 3340/3344 track format used with System/3 includes 2 bytes in each count area and home address (physical address, or PA) that are used for seek verification. Whenever the attachment logic processes the home address to perform an operation, the PA bytes from the disk are compared with PA bytes generated by logic from the most recent seek address. A noncompare condition results in the operation ending at this point with seek check status indicated.

3340 Data Privacy

Data privacy is a programmer responsibility on a 3340 attached to the System/3. The setting of the 3348 read-only switch can be checked at any time by issuing a read diagnostic sense command.

3340/3344 TRACK FORMAT

Figure 6-23 shows the 3340/3344 track format. The format for each halftrack written on the data module starts at a point on the disk called the index marker. This point is specified by a signal emitted by the disk spindle as it turns, generating synchronized index markers for each recording surface of all disks in the module. The drive provides two index markers per recording surface, one at the start of the track, and another on the opposite side of the track (180° removed from the first index marker). A home address, then record 0 (a track identifier record), then sequentially numbered records follow the index marker on each halftrack until the next index marker is encountered, signalling that the entire halftrack has been used. Records following record 0 on the odd halftrack continue sequential numbering with a value 1 greater than the number of the last record on the even halftrack.

3340/3344 Index Markers

The even index marker signals the initial point of both the full track and the even halftrack and the final point of the odd halftrack. The odd index marker signals the final point of the even halftrack and the initial point of the odd halftrack.

The index marker is not recorded on the track or in storage. However, it is shown in this manual as a track reference point.

3340/3344 Gaps

A gap is an area written on the track by attachment logic to separate two adjacent groups of information and to identify the group that follows the gap. Gaps are used by attachment logic; they are never used by the system program.

Tracks can be initialized and records written in either of two formats: standard data format or compressed data format. IBM System/3 programming support always uses the compressed data format on normal data tracks for 3340/3344 programs. These two formats differ from each other in the number of count areas used per halftrack and the key and data area specifications. The two formats cannot be intermixed on any halftrack.

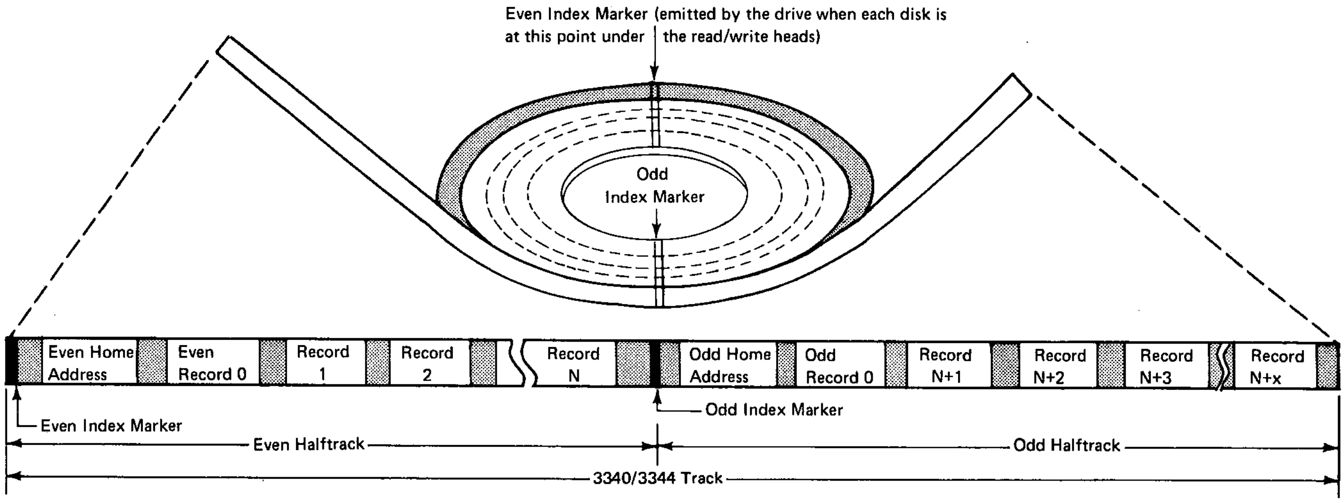
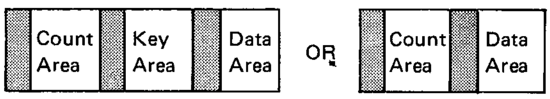


Figure 7-24. 3340/3344 Track Format

3340/3344 Standard Data Format

As just illustrated, tracks formatted in the standard data format have a variable number of records (51 maximum, including R0) per halftrack. The HA field on each halftrack is the same as described earlier in this section. Numbered records, including R0, are formatted as described below. Note that each numbered record has an unshared count area:

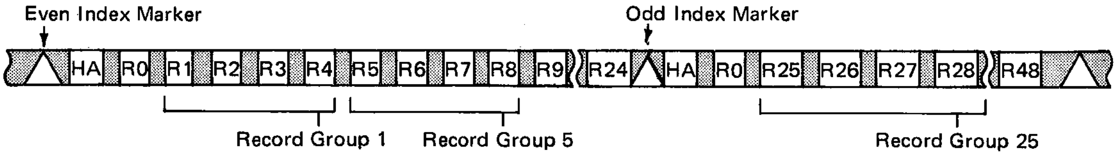


In this format, the count area KL byte specified a key length greater than hex 00.

In this format, the count area KL byte specified a key length of hex 00.

The KL specification can vary from record to record when the program uses the standard data format. The number of records per halftrack is a function of the format used and the number of key bytes plus data bytes per record.

3340/3344 Compressed Data Format

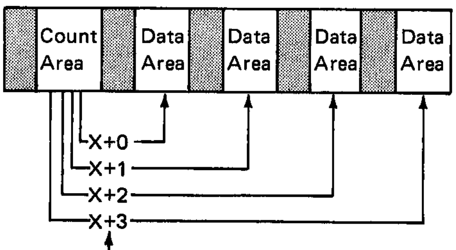


The preceding drawing shows the format of a track written in compressed data format. In this format, each halftrack has an HA, an R0, and 24 sequentially numbered records. The HA is exactly as described earlier in this section. The R0 is a standard record format record with a key length of hex 0 and a data length of 08; therefore, R0 contains a count area and a data area, but no key area.

R1 through R24 on the even halftrack and R25 through R48 on the odd halftrack are divided into 12 compressed record groups of four adjacent records each. To save track space, each record has a key length of hex 0 (and therefore no key area) and a standard data length of 256 bytes.

Records are identified by track number and record number, and the track condition is indicated by bits 6 and 7 of the flag byte in the count area. For each compressed record group, the attachment provides a single (common) count area followed by four data areas. This count area contains the address of the first record in the group, and the attachment uses this number as a base number from which it can apply a displacement value of 0, 1, 2, or 3 to identify the first, second, third, and fourth records. The track number and track condition apply for all records in the group.

The following drawing represents a compressed record group:

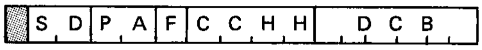


Record number, where X is the value stored in the R-byte of the count area. Valid X values in the count areas are 1, 5, 9, 13, 17, 21, 25, 29, 33, 37, 41, and 45.

Compressed record groups can be identified by the number of the first record in the group, which is the number stored in the group's count area R-byte (Figure 7-25).

Intermixing record formats on any track (that is, standard format on the even halftrack and compressed format on the odd halftrack, or vice versa) is a poor programming practice because the program could then assign the same record number to two different records on the track—one on the even halftrack and one on the odd halftrack). If record formats are to be intermixed for an application, full tracks should be formatted using the same type of format.

3340/3344 HA (Home Address)



Each halftrack contains a 15-byte home address that identifies its physical location (PA), logical address (CCHH), and condition (F). The even home address is the first recorded area following an even index marker, and the odd home address is the first recorded area following an odd index marker.

Home addresses are written at the IBM disk manufacturing plant. They are rewritten during a mandatory reinitialization to format the disk for IBM System/3 programming. The home address area of each halftrack on a defective track and on each halftrack of an assigned alternate track are also rewritten during alternate track assignment procedures.

Gap G1, which always follows an index marker, precedes the first byte of the home address. The home address field is always followed by a G2 gap, which separates the home address from R0, the track descriptor record.

The following commands are used for writing and reading a home address area:

Write HA and R0 count even
Write HA and R0 count odd
Read HA and R0 count even
Read HA and R0 count odd

3340/3344 SD (Skip Displacement) Field in Home Address

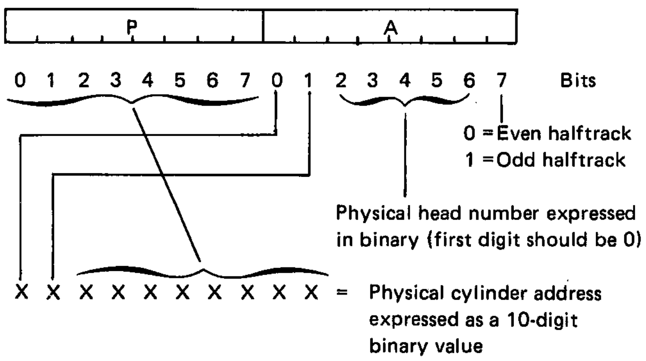
This field can identify a bad area on the halftrack. If the field contains hex 0000, there is no defect; otherwise, the value written in the field indicates the distance (in bytes) from the halftrack index marker to the center of a defect. Although the SD field is not used for normal programming, it can be recovered by issuing the appropriate read HA and R0 count instruction, with no resulting unit check, then issuing a read diagnostic sense instruction.

Count Area R-Byte Contents[1]			Identification of Records in Compressed Record Group[1]							
			First Record		Second Record		Third Record		Fourth Record	
Dec	Binary	Hex	Dec	Hex	Dec	Hex	Dec	Hex	Dec	Hex
1	0000 0001	01	1	01	2	02	3	03	4	04
5	0000 0101	05	5	05	6	06	7	07	8	08
9	0000 1001	09	9	09	10	0A	11	0B	12	0C
13	0000 1101	0D	13	0D	14	0E	15	0F	16	10
17	0001 0001	11	17	11	18	12	19	13	20	14
21	0001 0101	15	21	15	22	16	23	17	24	18
25	0001 1001	19	25	19	26	1A	27	1B	28	1C
29	0001 1101	1D	29	1D	30	1E	31	1F	32	20
33	0010 0001	21	33	21	34	22	35	23	36	24
37	0010 0101	25	37	25	38	26	39	27	40	28
41	0010 1001	29	41	29	42	2A	43	2B	44	2C
45	0010 1101	2D	45	2D	46	2E	47	2F	48	30

[1]The record number stored in the count area R-byte (the number of the first record in the compressed record group) identifies the compressed record group. For example, R7 is in group 5 (decimal) and R40 is in group 37 (decimal).

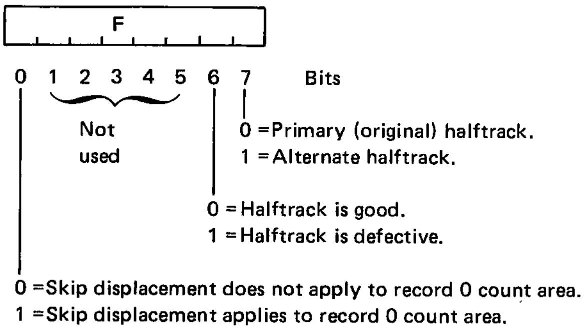
Figure 7-25. Compressed Record Group Summary

3340/3344 PA (Physical Address) Field in Home Address



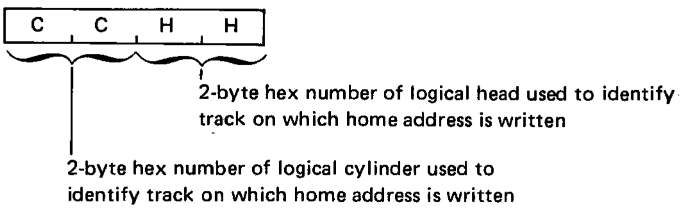
The two PA bytes are written by the manufacturing plant to identify the actual track being used. The adapter reads the actual PA bytes from the drive whenever a home address and record 0 is read and compares them with PA bytes generated by adapter logic for seek verification. These bytes are not available for normal programming.

3340/3344 F (Flag) Byte in Home Address



Although this field identifies the condition and use of the halftrack in which it resides, both home address flag bytes in the track must be changed when either home address flag byte is changed.

3340/3344 CCHH (Track ID) Field in Home Address

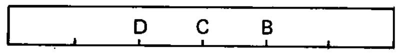


If the system program issues an instruction that contains a cylinder number greater than hex 00D1 (decimal 209), a head number greater than hex 0013 (decimal 19) for cylinders hex 0000 through 00D0, or a head number greater than hex 0007 (3340) or hex 0013 (3344) for cylinder hex 00D1, the attachment posts a command reject indication.

All bits of these 4 bytes have identical meanings for both the 3340 and 3344, except bits 0 and 1 of the leftmost C-byte, where:

Bits 0 and 1	Meaning
0 0	Logical volume 1
0 1	Logical volume 2
1 0	Logical volume 3
1 1	Logical volume 4

3340/3344 DCB (Detection Code Bytes) Field in Home Address



The 6 DCB bytes are generated and used by the controller for error detection. These bytes enable the controller to detect all single error bursts in a span of fewer than 12 bits.

The home address, count area, and key area all have DCB fields. The DCB fields in the count and key areas serve identical functions to the home address DCB field function.

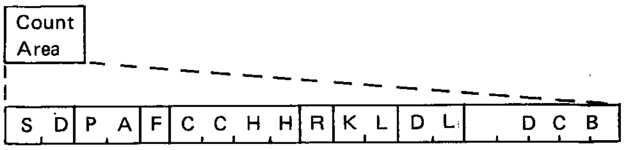
3340/3344 Records (R0, R1, R2, . . .)

Numbered records fill the track from the home address of each halftrack to the end of that halftrack. The first numbered record on the even halftrack is even R0, and all subsequent records are numbered sequentially, from R1 through Rn (that last numbered record on the even half-track). Rn is followed by gap G3, the odd index marker, gap G1, the odd home address, gap G3, the odd R0, then the next sequentially numbered record on the track, Rn+1. Note that sequential numbering interrupted by the odd index marker, odd home address, and odd R0 resumes with this record from the last sequentially numbered record on the even halftrack (Figure 7-23).

A numbered record always contains a count area and a data area. A record can also contain a key area between the count and data areas.

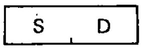
IBM System/3 programming support uses R0 count areas to identify alternate tracks when the primary tracks be-come defective.

3340/3344 Record Count Area



The count area identifies the record and defines the number of bytes in the key and data areas of the record. During record operations, the attachment compares the record identification data (bits 6 and 7 of the flag byte, the cylinder number, the head number, and the record number) from the disk drive control field in processing unit storage with the corresponding fields written in the count area of a record passing the read/write head. A compare equal condition indicates that the desired record is passing the head: this is called *record orientation.* If record orientation does not occur, within 1-1/2 revolutions of the disk, attachment logic posts a no-record-found indication.

3340/3344 Count Area Skip Displacement Field (SD)

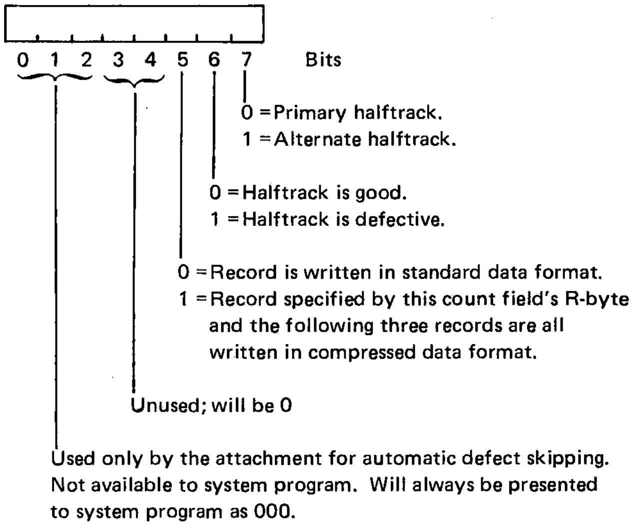


This field is used by the attachment to identify and bypass a defective area on the halftrack. The field is not available to the system program.

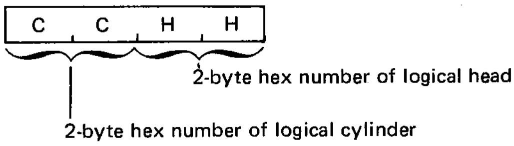
3340/3344 Count Area Physical Address Field (PA)

This field, which is used by the attachment to identify each physical track on the data module, is identical to the home address PA field.

3340/3344 Count Area Flag Byte (F)



3340/3344 Count Area Track ID Field (CCHH)



The logical head number and logical cylinder number identify the logical track number assigned to the physical track on which the record is recorded. When the track number is used with the record number, the system can locate any record on the entire data module. See note under *3340/3344 CCHH (Track ID) Field in Home Address* for 3340/3344 differences.

3340/3344 Count Area Record Number Byte (R)

This single byte field is used to identify the record (in standard data format) or records (in compressed data format) associated with the count field.

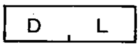
For standard data format operations, the record number byte contains the number of the record in which the count field resides. For compressed data format operations, the byte serves as a base number to which the attachment can add a displacement value of 0, 1, 2, or 3 to identify any one of the four records that share the same count area.

When used with the CCHH (track ID) bytes, the record bytes identifies any record or compressed record group in the data module.

3340/3344 Count Area Key Length Byte (KL)

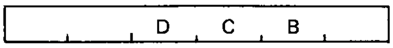
The key length byte is used only in the count area. It specifies the number of data bytes in the key area of the record. Valid key lengths are 0 through 255 decimal, or 00 through FF hex. However, in System/3, the key length is also conditioned by the data length specified, because the total value of the key area plus the data area on the record cannot exceed 256 bytes (decimal). For those installations using IBM System/3 programming support, the key length must be 0 for normal data tracks.

3340/3344 Count Area Data Length Field (DL)



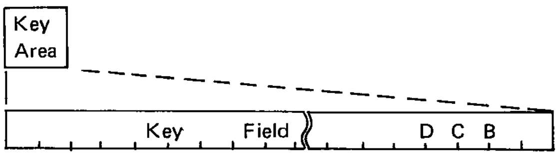
This 2-byte field is used only in the count area. It specifies the number of data bytes in the data area of the record. Valid data lengths are 0 through 256 decimal, or 0000 through 0100 hex. However, in System/3 the data length is also conditioned by the key length specified, because the total value of the data area plus the key area on the record cannot exceed 256 bytes (decimal). For those installations using IBM System/3 programming support, the data length of both even and odd R0 is always 8, and the data length of all other records is always 256 decimal (0100 hex) for normal data tracks.

3340/3344 Count Area Detection Code Bytes (DCB)



This field's function is identical to that of the home address DCB field. It is used by the controller for error detection in the count area.

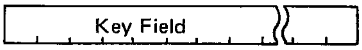
3340/3344 Record Key Area



The key area is never in the record format unless the KL (key length field) in the count area specifies a value greater than hex 00. When this field is used in the format, it follows the count area and its trailing gap.

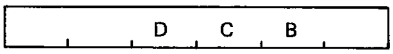
The key area is not used in R0 or in any compressed format record.

3340/3344 Key Area Key Field



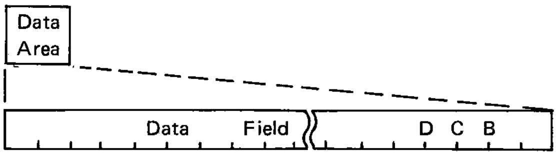
The key field can contain from 1 through 255 bytes of data. It usually contains record identifying information such as serial number or policy number. The number of key bytes in the key field is specified by the key length byte in the last preceding count area. The key field and the data field in any single record cannot total more than 256 bytes. Compressed format records have no key field.

3340/3344 Key Area Detection Code Bytes (DCB)



This field, which is identical in function to the home address DCB field, is used by the controller for error detection in the key area.

3340/3344 Record Data Area

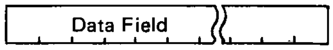


Every numbered record has a data area. In compressed record format, the four data areas following a count area on the track share that count area; that is, the count area preceding any data area on a track always provides count information for that data area.

With standard formatting, the data area follows the key field if the record has a key length greater than hex 00, and follows the count field if the record has a key length of hex 00.

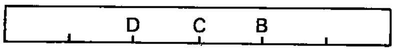
With compressed record formatting, the data field either follows the count field (for records 1, 5, 9, 13, and every succeeding fourth record on the track) or follows another data field.

3340/3344 Data Area Data Field



The data field can contain from 1 through 256 bytes of data. The number of bytes in the data field is specified by the DL field in the count area in the last preceding count area. The key field and data field in any single record cannot total more than 256 bytes. For systems using IBM System/3 programming support, the data field for R0 always contains 8 bytes and the data field for each other numbered record on every normal data track always contains 256 bytes.

3340/3344 Data Area Detection Code Bytes (DCB)



The controller generates and uses this 6-byte DCB field, which provides an error detection function identical to the error detection function provided by the home address and count area DCB fields. The controller also uses this field to regenerate data written in bad spots on the disk that do not exceed 3 adjacent bits on the track. When data regeneration is accomplished successfully, the controller passes the corrected bits to the attachment as valid data. Otherwise, the controller indicates a data error.

LOCAL STORAGE REGISTERS USED FOR 3340/3344 PROGRAMMING

3340/3344 Disk Drive Data Address Register (DDDR)

The DDDR contains the 2-byte address of the 3340/3344 data field (DDDF). Whenever the system is to provide data to the drive (for example, for a write or scan operation) or is to receive data from the drive (as during a read operation) the DDDR must contain the address of the leftmost byte of the data field.

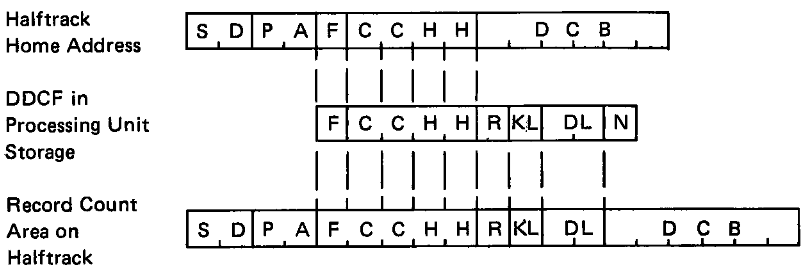
3340/3344 Disk Drive Control Register (DDCR)

The DDCR contains the 2-byte address of the 3340/3344 disk drive control field (DDCF). The DDCR must contain the address of the leftmost byte of the DDCF at the start of any SIO operation that uses the contents of the DDCF.

3340/3344 Disk Drive Control Field (DDCF)

F	C C	H H	R	KL	DL	N

The disk drive control field is a system-program-defined field in main storage that contains a 10-byte control argument for most start I/O instructions. The DDCF can start on any byte boundary addressed by the disk drive control register (DDCR). As shown below, all the bytes except the N-byte in the DDCF have similar bytes in the record count areas on the disk. The DDCF cylinder number and head number bytes (used to identify logical tracks on the data module) have related bytes in both the home address areas and the record count areas on every halftrack in the data module.



The system program generally preloads the defined DDCF with the control argument for the operation before issuing a disk-related start I/O command. Program modification of the DDCF must not be attempted while the disk drive attachment is busy. The functional significance of each DDCF byte except the R-byte and the N-byte is identical to that of the corresponding byte in the record count area.

3340/3344 DDCF R-Byte

R

This byte specifies the sequential number of the record on the track. Valid record numbers are 0 through 255 decimal (00 through FF hex). The R-byte must match the corresponding byte in the disk count area before record orientation can occur. Although the R-byte can specify a valid number of up to 255, the track cannot contain any record numbered greater than decimal 100.

3340/3344 DDCF N-Byte

N

This byte specifies the number of additional fixed format records to be operated on. Therefore, a control field with an N-byte of hex 5 specifies an operation on the addressed record and the following five records. (Such an operation is called a multiple record operation.) Valid N-bytes range from decimal 0 through 255.

3340/3344 MULTIPLE FIXED FORMAT RECORDS

Multiple fixed format records are defined as sequential records having equal length key areas and equal length data areas. A control field with an N-byte specifying other than 0 causes multiple fixed length records to be operated on. (*Exception:* If R0 is specified by the R-byte as the starting record, only record 0 is operated on, regardless of the content of the R-byte.) For the write count compressed data command, the record number (n+1) must be evenly divisible by 4, or command reject status is posted.

At the start of the operation, the attachment writes the contents of the main storage DDCF into attachment control storage. If the R-byte contains a 0, the attachment loads a 0 into its N-byte, overlaying whatever value was transferred from main storage. The attachment then operates on the first record. After the record has been successfully operated on, the attachment:

1. Decrements the N-byte value by 1. (Decrementing by 1 from an N-byte of 0 places a hex FF in the N-byte.
2. Examines the contents of the N-byte. If the N-byte value is hex FF, there are no more records to be operated on and the operation ends. If the N-byte contains other than hex FF, the value contained in the N-byte, plus 1, specifies how many more records must be operated on, and the attachment continues with the operation.
3. Increments the DDCF R-byte by 1 to specify the nex sequential record as the record to be operated on.
4. Performs the operation specified by the instruction on the record specified by the updated R-byte.
5. Transfers the residual DDCF from the attachment into the main storage DDCF at the successful completion of the operation.

Note: These functions are slightly modified if head switching occurs during the operation.

3340/3344 HEAD SWITCHING AND CYLINDER SWITCHING

During multiple-record operations, a single start I/O instruction can cause as many as 256 records to be operated upon (the record specified by the R-byte, plus another 255 identically formatted records trailing the specified record in the disk drive, as specified by an N-byte of hex FF in the DDCF). In many cases, some of the multiple records must be read from the originally specified track, and the next records must be read from a second track. To operate on records from two different tracks as the result of a single instruction, the attachment switches read/write heads, switching from the presently active logical head to the next higher numbered logical head after the last record on the original track has been operated upon. During head-switching, the attachment increments the logical head number by 1 and resets the record number to 1. Therefore, the next record operated on is record 1 (note that record 0 is bypassed) of the newly selected track.

Whenever the attachment recognizes the end of any logical cylinder from hex 0000 through 00D0 (X0D1[1] on 3344), it increments the logical cylinder number by 1, resets the logical head number to 0, and resets the record number to 1. In this case, the next record operated on is record 1 of the first track in the next sequential logical cylinder.

On 3348-70 data modules, head switching does not occur from cylinder 209 decimal (hex 00D1) head 7 decimal (hex 0007). On 3344 data storage, head switching does not occur from cylinder 209 decimal (hex X0D1[1]) head 19 decimal (hex 16). If a multiple record operation exceeds either of these points, the attachment returns an end-of-cylinder indication to show that the data area on the data storage has been exceeded.

If the head switches to a track that has been flagged as defective, the attachment posts a unit check with a track condition indication and stops the operation.

3340/3344 RESIDUAL VALUES

The residual contents of the DDCF, DDCR, DDDF, and DDDR are defined for normal ending status conditions following the issuance of a start I/O command specifying the 3340. If a unit check is posted, the residual value of the DDCF is defined for format 0, message D (defective track error) and format 0, message E (alternate track error) only. If format 0, message D is posted, the CCHH points to the defective track, and R indicates the record that the attachment was trying to process. If format 0 message E is posted, the CCHHR points to R1 of the defective track plus 1, assuming the CCHH of the defective track was written on the alternate track.

[1]All bits of these 4 bytes have identical meanings for both the 3340 and the 3344, except bits 0 and 1 of the leftmost C-byte, where:

Bits 0 and 1	Meaning
0 0	Logical volume 1
0 1	Logical volume 2
1 0	Logical volume 3
1 1	Logical volume 4

If any unit check is posted without adapter check, the residual DDCF is valid and defines the current count field or HA field as calculated by the adapter. To restate, if a data check has occurred on read HA or count field, the residual DDCF contains calculated values rather than the actual DDCF read from disk. However, the appropriate diagnostic sense bytes reflect actual DDCF values read from disk. This allows programming recovery from data checks with minimum DDCF restoration by programming.

Normal residual values for the DDCF, DDCR, DDDF, and DDDR are defined with the writeup discussing each start I/O command. In all cases, DDCF is copied into the attachment prior to command execution and is updated in the attachment during command execution. The residual value is stored back into the DDCF at the end of command execution.

3340/3344 DDCF Updating

- CC—The cylinder number is incremented by 1 at the end of a System/3 logical cylinder, provided the conditions listed under HH exist. If the residual CC is posted for a 3344 command, the logical volume is also posted in the CC.

- HH—The head number is incremented by 1, causing head switching, at the even index marker if all these conditions apply:

  1. Record orientation has occurred.

  2. The last record for the current command has not been processed.

  3. Check status has not been posted.

  4. HH is less than hex 13 (if equal to hex 13, HH is reset to 0).

- R—The record number is incremented by 1 at the end of each data field if:

  1. Record orientation has occurred.

  2. The last record for the current command has not been processed.

  3. Check status has not been posted.

  4. Even index has not been detected (reset to 1 if even index was detected).

- N—The N-byte, which indicates the number of records remaining to be processed minus 1, is decremented by 1 at the end of each data field if record orientation has occurred.

If any check status is posted without attachment check status, the CCHHR bytes contain the address of the last record the attachment tried to process.

If attachment check status is posted, the content of the control field cannot be considered valid.

3340/3344 DDCR Residuals

The DDCR is returned to its initialized value at the end of any operation in which it is used. If an attachment check is posted, the contents of the register cannot be considered valid.

3340/3344 DDDF Residuals

The residual contents of the DDDF at the end of start I/O disk operations is discussed with the write-up describing the SIO operations.

If a data check status is posted without adapter check status and read HA R0 even/odd command was issued, the residual DDDF is not transferred to main storage, and the DDDF remains as initially defined.

If a unit check occurs during a SCAN command at a time during data transfer (after SCAN hit) from disk to attachment buffer, the transfer of the residual DDDF to main storage does not occur, and the initial SCAN argument is preserved in main storage.

3340/3344 DDDR Residuals

If an adapter check status is posted, the DDDR contents cannot be considered valid. The DDDR must be reinitialized during error recovery procedures. If any check status is posted without adapter check, the DDDR contains the address of the last byte stored plus 1.

If data overrun status is posted, the DDDR contains the address of the last DDDR position operated on.

3340/3344 IN-PROCESS CONDITIONS

- A test I/O and branch instruction is never rejected.

- Start I/O control, read, write, and scan instructions are rejected whenever the adapter busy indicator is on.

- Start I/O interrupt is never rejected.

- Load disk drive data register and load disk drive control register instructions are rejected if the adapter busy indicator is on and the adapter microprocessor clock is running.

- Load I/O diagnostic is never rejected.

- Sense I/O is never rejected.

- All SIOs except read sense and interrupt control type SIOs are rejected when DM attention indicators are on.

The system loops on rejected instructions until the attachment logic accepts them. The attachment logic accepts start I/O instructions provisionally, executing them when the attachment and drive have completed any operations in process.

If the program issues a start-I/O instruction for a not-ready drive, the attachment accepts the SIO, and posts attachment busy and I/O attention; a subsequent test I/O for not-ready/unit check results in a condition met response. The attachment remains busy until the drive becomes ready and the SIO has been executed. I/O attention drops when the drive becomes ready. Attachment busy drops when execution of the SIO ends.

The preceding action has two exceptions:

1. If the program issues a read-and-reset-buffered-log, read-diagnostic-sense, read-extended-functional-sense, or read-and-reset-data-module-attention-interrupt-control instruction, the attachment logic does not set the I/O attention indicator. Instead, the attachment executes the SIO immediately, dropping the attachment busy indication when the operation ends.

2. If the TIO for not-ready/unit check returns a condition not met response (that is, if the drive is ready) and the addressed drive goes not ready before the attachment can execute the SIO, the attachment aborts the SIO instruction, sets the unit check indicator and no-op status bit, and requests an op-end interrupt.

If the program issues a start I/O to a 3340 on a Model 12, when (1) the data module is in the read-only mode and the SIO specifies a write operation, or (2) the installed data module is not a 3348-70 module, the attachment accepts the SIO and posts attachment busy. Subsequently, the attachment turns I/O ATTENTION on and posts unit check. I/O ATTENTION turns off when the drive becomes ready. At this time the attachment posts no-op, sets the op-end indicator (if enabled), and drops busy.

3340/3344 NO-OP CONDITIONS

A no-op condition occurs when the attachment accepts an instruction but cannot execute it for some reason. Whenever this condition occurs, the attachment sets the no-op attachment sense bit (sense byte 1, bit 4) and requests an op-end interrupt. Possible reasons for a no-op condition are:

1. The program issues an SIO instruction other than recalibrate, read diagnostic sense, data module attention control reset, or read extended functional sense to a disk drive with an outstanding seek incomplete status.

2. The drive cannot execute a provisionally accepted SIO instruction because the drive has an outstanding seek incomplete status.

3. The program issues an SIO requiring drive activity and the drive went not-ready while the attachment was processing the instruction.

4. The program issues a write command to a drive and writing is inhibited because the data module slide is set at its READ ONLY position.

5. The program issues a seek, recalibrate, read, scan, or write instruction while the disk drive control field contained an invalid track address in its CCHH bytes, or an invalid record number in its R-byte.

6. The program issues a write HA and R0 instruction before issuing a read HA and R0 instruction to the same halftrack.

7. The program issues an instruction that attempts to process any record except R0 on a track that has been flagged defective. (HA areas can be processed.)

8. The program issues a write HA and R0, a write count key data, a write R0 odd, a write count compressed key data, a write key data, a write repeat key data, a read count key data diagnostic, or a read key data command while the disk drive control field specifies key length and data length with a combined total greater than decimal 256.

9. The program issues a start I/O instruction with a Q-byte and R-byte that do not represent a valid instruction.

10. The program issues an SIO instruction other than read-and-reset-buffered-log, read-diagnostic-sense, data-module-attention-control-reset, or read-extended-functional-sense to a disk drive with an outstanding data module attention condition.

3340/3344 TIMING

The total access and data transfer time is the sum of the time required for access motion, head selection, rotational delay, and data transfer.

3340/3344 Access Motion Time

Access motion time is the time required to position the access mechanism at the specified cylinder. If the access mechanism is already at the proper cylinder, access motion time is 0. However, if the access mechanism is moved, the following times are required:

Parameters	Time in Milliseconds
Minimum: (1 access position)	10
Average (random):	25
Maximum: (350 access positions on 3348-70, and 560 access positions on 3344)	50

3340/3344 Head Selection Time

The time required to select the read/write head is negligible.

3340/3344 Rotational Delay Time

Rotational delay is the time required for the desired record area to reach the read/write head so that data transfer can begin. This time can range from slightly more than 0 to over a full revolution. The maximum and average rotational delays for 3340/3344 drives are:

Maximum rotational delay:	21.5 milliseconds
Average rotational delay:	10.8 milliseconds

3340/3344 Data Transfer Time

Nominal read/write rates for the disk drives are:

Bytes per second:	885,000
Microseconds per byte:	1.13

3340/3344 OPERATIONS

Preparing a 3340/3344 for Initial Operation

A 3340/3344 cannot operate on an IBM System/3 until a data module initialized for System/3 use has been mounted on the 3340 and the attachment initial microprogram load (IMPL) routine has been performed. This routine stores micro instructions in the attachment control storage area and initializes the attachment for system operation.

The customer is responsible for having the storage devices initialized so that initial microprogram load programs and the functional attachment microprogram can be loaded on the storage devices. Data modules prepared for System/3 by the IBM Program Library (PID) contain these microprograms. Also, the customer engineer can load the microprogram when the system is installed.

When IBM programming support is being used, the initial program resides on disk or an alternate medium, such as cards. The initial program load procedure is:

1. Make the system ready for operation.

2. Set the PROGRAM LOAD SELECTOR switch to the appropriate setting.

3. Press the PROGRAM LOAD key.

When IBM programming support is not being used, the following steps build a data module that contains the routine used to initialize the attachment and loads the attachment functional microprogram into attachment control storage:

1. Load the 3340/3344 function microcode.
   a. Load the following programs (provided with CE diagnostic package) from the alternate loading device in the following sequence: LDS, FA0.
   b. Set the PROGRAM LOAD SELECTOR switch to ALTERNATE.
   c. Press PROGRAM LOAD.

   Note: The following cylinder locations must be available for use on each 3348 data module that is used for initial microprogram loading, and the tracks must be initialized in compressed data format:

   – Cylinder 0, head 0, records 25 through 29 (to store FA6 and FA7)
   – Cylinder 0, head 0, records 33 through 37 (to store FA6 and FA7)
   – Cylinder 0, head 0, record 46 (to store the control program link–3344)
   – Cylinder 0, head 0, record 47 (to store EC level of FA6, FA7, and FA0)
   – Cylinder 0, head 0, record 48 (to store the control program link–3348-70)
   – Cylinder 0, head 2, (or its assigned alternate) records 1 through 48 (to store FA0)

2. If the customer intends to use the IBM initializing program for the system, the IBM programs needed to make the system fully operational and to initialize the system from disk can be loaded with the following procedure:
   a. Load the following programs (provided with CE diagnostic package) from the alternate loading device in the following sequence: FFF, 143 and FC0 (Model 15 only), FC2, FA6, FA7, C17, FA0. Data from these programs will be placed on the initialized modules.
   b. Set the PROGRAM LOAD SELECTOR switch to ALTERNATE.
   c. Press PROGRAM LOAD.

When the SCP supplied from PID is used, the module shipped with your order will contain a programming system that can be used to load the initial program into the system.

The $INIT utility program formats other modules to be used by System/3 and updates initial program load records on the module being initialized. The $SCOPY utility program can be used to update the IPL records to the latest level on modules already initialized.

3340/3344 Seek Operation

The seek control command selects one of the logical primary or one of the logical alternate tracks on the disk drive specified by the DA- and M-code portions of the Q-byte. After a seek operation, a *logical cylinder* remains selected until a different cylinder is selected by a subsequent seek or recalibrate operation or until automatic cylinder switching occurs. A *track* remains selected until a different track is selected by a new seek or recalibrate operation or until automatic head switching occurs. (A *track* that is initially selected by a seek or recalibrate operation is changed by any subsequent multiple-record read, write, or scan command that causes automatic head switching to occur. Cylinder overflow from the last logical track on one logical cylinder to the first logical track on the next higher numbered logical cylinder occurs during multiple-record operations on all except logical cylinder 209 (decimal). At the end of track 7, logical cylinder 209, the adapter sets end of cylinder if the operation requires additional records. The seek command does not verify that the correct track was selected, but the attachment performs invalid head number and cylinder number checking. Command reject occurs if the seek argument is greater than cylinder 209, head 7 (decimal) or the head value is greater than decimal 19.

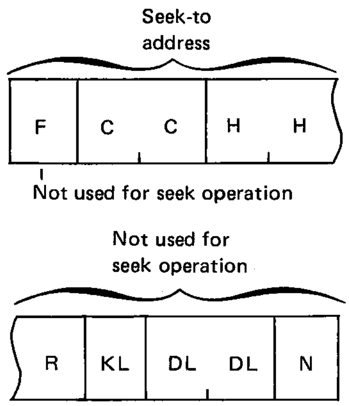
The system program can initiate a zero cylinder seek (that is, a seek to the same cylinder) to perform head selection. The attachment provisionally accepts (stacks) a read, write, or scan command to any drive while a seek or recalibrate command is being executed on that drive. The stacked command is executed immediately at the end of the seek or recalibrate operation if seek check or adapter check status is not posted. If a seek check is posted, the system program must issue a recalibrate command to clear the seek check in the drive.

If the program issues a seek or recalibrate command to a drive that is currently executing a seek or recalibrate command, the attachment may set the command reject and no-op status bits, then post a seek complete interrupt/op-end indication. Therefore, the program should never issue two consecutive seeks to the same drive without a programmed intervening check for a seek complete condition (via TIO or interrupt).

Drive seek complete status (byte 0, bits 4-7) is available only if 3340/3344 interrupts/op-end indicators have been enabled.

Initial Conditions

DDCF—Contains the 4-byte seek address (CCHH) used to specify the seek to cylinder and head number. The remaining bytes in the DDCF are not used:



DDCR—Must contain the address of the high-order (left-most) byte of the DDCF.

DDDF—Unchanged.

DDDR—Unchanged.

In Process Conditions

Test I/O—Selected device seek-busy response is positive until the seek operation is completed.

Test I/O—Attachment busy is positive:

1. From the time the seek command is issued until the drive accepts the seek information

2. From provisional acceptance of a read, write, or scan command until the operation is completed

An overlapped seek operation can be initiated if the selected device is not seek busy or attachment busy.

Ending Conditions

DDCF—Remains unchanged.

DDCR—Contains the initialized address. The contents of the register are unpredictable if attachment check status is posted.

3340/3344 Recalibrate Operation

The recalibrate control command starts a direct seek to cylinder 0 and head 0. Execution is the same as that for the seek command except for command execution times. Initial control and register fields need not be specified and remain unchanged.

Note: Use an SIO recalibrate instruction to reset a detected seek incomplete condition.

3340/3344 Read Home Address and Record 0 Count Even Operation

This instruction transfers the 5-byte home address field (FCCHH) into the main storage disk drive control field, and the 9-byte record 0 count field (CCHHRKLDLDLN) into the disk drive data field from the even halftrack passing the read/write head.

Initial Conditions

DDCF—Destination field for information from the first 5 bytes (FCCHH) of the home address area of the halftrack. (These bytes hold the flag data and the track address.)

DDCR—Must contain the address of the leftmost byte of the DDCF.

DDDF—Destination field for record 0 (R0) count field bytes from the halftrack.

DDDR—Must contain the address of the leftmost byte of the DDDF.

Ending Conditions

DDCF—Contains the flag byte and track number from the home address area of the track.

DDCR—Contains the initial address. (If an equipment check is posted, the contents of the register are not guaranteed.)

DDDF—Contains record 0 count field bytes.

DDDR—Contains the starting DDDR value, plus 9.

3340/3344 Read Home Address and Record 0 Count Odd Operation

The read HA and R0 count odd instruction is identical to the read HA and R0 count even instruction except that for the read HA and R0 count odd instruction, information is from the odd halftrack.

3340/3344 Read Record 0 Key Data Odd Operation

The read R0 key data instruction transfers the key and data fields from the odd record 0 (the record 0 past the odd index) into the disk drive data field. This instruction is similar to the read key data instruction with R = 00 (that is, specifying record 0 as the record to be read). However, in this instruction, the R-byte in the disk drive control field is not used and the data is read from the odd halftrack. The N-byte in the disk drive control field must be 00.

3340/3344 Read Key Data Operation

The read key data operation transfers one or more disk records from the selected 3340/3344 track into main storage. Reading begins at the record specified by the identifier field (CCHHR) in the disk drive control field in main storage. Record orientation is conditioned (that is, the correct record is assumed to have been found on the track) when flag byte bits 6 and 7 and the identifier field of a record on the disk track exactly match those fields in the disk drive control field (DDCF) located in main storage.

If record orientation does not occur within one and one-half revolutions of the disk, the attachment logic posts a no-record-found indication. Attachment logic uses the key and data lengths from the DDCF to determine how many bytes to read from the disk, ignoring the key and data lengths on the record count field on the disk. However, if the DDCF specifies an incorrect key length or data length, a data check may occur.

When properly specified by the disk drive control field, R0 on a track can be read. However, the drive bypasses R0 whenever R0 is encountered after head switching during multiple-record operation. During head switching operations, the attachment selects record 1 on the next sequential track as the next record to be read, thereby bypassing R0.

Note: Head switching occurs at even index time if record orientation was successful and the last record in a multiple record operation has not been read. Cylinder switching occurs at even index time after the last track on any logical cylinder except decimal 209 has been read. On logical cylinder 209, the attachment posts an end-of-cylinder indication after track 7 has been read.

Initial Conditions

DDCF—Must contain the starting disk record address.

DDCR—Must contain the address of the leftmost byte of the DDCF located in main storage.

DDDF—Main storage area to receive the contiguous key and data fields from disk storage. Field length = (N+1) (key length + data length).

DDDR—Must contain the address of the leftmost byte of the DDDF.

Ending Conditions

DDCF—Identifier portion contains the address of the last record read. The N-byte portion residual equals hex FF if all records were read.

DDCR—Contains the initialized address of the DDCF.

DDDF—Contains contiguous key and data fields read from the disk.

DDDR—Contains the address of the last DDDF location operated on, plus 1. That is, disk drive data record 0 + (N+1) (key length + data length) where the disk drive data record 0 is the initialized contents.

3340/3344 Read Count Key Data Operation

This instruction recovers a single record when the key and data lengths of the record to be read are not known.

The attachment starts the operation by orienting on the record specified by the FCCHHR bytes in the DDCF. After orientation, the attachment transfers the 9 bytes from the count area on the record from the disk into the DDCF in main storage. Then, using the key and data lengths extracted from the Rn count area, the attachment transfers the contents of the key and data fields from the disk record into the DDDF.

Even R0 can be specified as the record to be read. In this case, orientation occurs on the home address area of the even halftrack. Odd R0 cannot be read without issuing a read R0 key data odd or a read HA and R0 count odd instruction.

Initial Conditions

DDCF—Must specify the address of the disk record (Rn) to be recovered. Key and data lengths need not be specified. N must be 00.

DDCR—Must contain the address of the leftmost byte of the DDCF.

DDDF—CPU field that receives the contents of contiguous key and data fields from the disk drive. Field length is unknown to programmer and may be as large as 256 bytes.

DDDR—Must contain the address of the leftmost byte of the DDDF.

Ending Conditions

DDCF—Contains the 9-byte count area for the record read.

DDCR—Contains the initialized address of the leftmost byte of the DDCF.

DDDF—Contains contiguous key and data fields from the record read.

DDDR—Contains the address of the last DDDF location operated on.

3340/3344 Read Verify Key Data Operation

The read verify key data operation performs a read back check of the key and data fields. This operation is the same as a normal read key data operation, except that data transfer does not take place. The attachment performs the read back check by comparing detection code bytes generated for each field with those on the disk. The contents of the key and data fields are not compared.

To ensure that data was written accurately, issue a verify key data instruction immediately after any write command that modifies the key or data fields. Verification begins at the record specified by the identifier portion of the DDCF. The attachment reads the key length and data length from the count field of the record on the disk, so these lengths do not have to be supplied by the program. To verify multiple consecutive records, specify the number of records to be verified, plus 1, in the DDCF.

A maximum of 256 records can be verified without reissuing a new command.

Head switching can occur during command execution. However, during head switching operations, the drive starts examining records on the new disk at the index marker and searches until it encounters the record assigned hex 01 before it restarts the verification function. This means that record 0 is not verified.

If R0 is specified as the first record in a multiple-record operation, the attachment sets the N-field in the DDCF to 0 and reads even R0 without posting an error indication. This stops the operation immediately after even R0 has been verified.

Initial Conditions

DDCF—Must contain the address of the first record to be verified, and the number of records (N+1) to be verified.

DDCR—Must contain the address of the leftmost byte of the DDCF.

DDDF—Not used.

DDDR—Not used.

Ending Conditions

DDCF—Identifier portion contains the address of the last record verified. The N-byte portion contains hex FF if all records were verified.

DDCR—Contains the initialized leftmost byte address of the DDCF.

DDDF—Remains unchanged.

DDDR—Remains unchanged.

3340/3344 Read Count Key Data Diagnostic Operation

The read count key data diagnostic instruction can be used to recover data from a single record when the record has a defective count area. To perform this operation on a record (Rn), the attachment:

1. Orients the operation on the even index marker.

2. Searches for the count field ahead of the count field for record Rn.
   a. If record Rn was written in standard track format, the attachment searches for the count field in record Rn-1. (If Rn = record 0, the attachment searches for the home address for the same halftrack.)
   b. If record Rn was written in compressed track format, the attachment searches for the count field in the prior record group, unless Rn = R1, R2, R3, or R4. For R1 through R4, the attachment searches for the count field in the even home address area.

3. After finding the required count field (specified by step 2), the attachment skips to the end of that record (if the record is a home address or Rn was written in standard track format) or record group, and its associated trailing gap.

4. If the attachment next reads the odd index marker, it bypasses the following home address and record 0.

5. The next record area on the track should be the count area that was unreadable. The attachment, using the key and data length specifications from the DDCF, reads the specified number of bytes from the track into the DDDF. If the record is in standard format and the DDCF specifies a KL of 0, the first block of information after the bad count field is read as the data field. (This may be a key field.) The sum of KL + DL must not exceed decimal 256; if it does, the attachment posts a command reject status.

Program Note

Specifying a KL or DL greater than the actual KL or DL will probably cause a data check. However, specifying a greater value lets the program recover data beyond the end of the key field or data field on the addressed record. The first 6 bytes beyond the end of the key/data area will be detection code bytes for the addressed record.

Initial Conditions

DDCF—The CCHHR bytes must contain the address of the record to be read. The KL byte must contain the known (or assumed) length of the key area of the record, and the DL bytes must contain the known or assumed length of the data area of the record to be read. (If the record was written in compressed track format, the record has no key area and has a data area containing 256 bytes.) Bit 5 of the F-byte must be 0 for standard track format, or 1 for compressed track format.

DDCR—Must contain the address of the leftmost byte of the DDCF.

DDDF—Destination field for contiguous key area and data area bytes. Field length = KL + DL.

DDDR—Must contain the address of the leftmost byte of the DDDF.

Ending Conditions

DDCF—Remains unchanged.

DDCR—Contains the initialized address of the leftmost byte of the DDCF.

DDDF—Contains contiguous key area and data area bytes read from the addressed record. (If the DDCF specified a key length of 0, the attachment will have read the first key or data area after the count area from the record. If the DDCF specified a KL + DL DL greater than the total number of bytes in the record, the additional DDDF positions will contain bytes read from the following bytes on the disk; the first six of these bytes will be detection code bytes.)

DDDR—Contains the address of the last DDDF location used plus 1, or the initialized DDDR address plus KL plus DL.

3340/3344 Read Diagnostic Sense Operation

Issuing a read diagnostic sense instruction transfers 24 bytes of information from the attachment to the DDDF. The type of operation just performed determines what type of information the attachment transfers during a read diagnostic sense operation:

1. If a sense instruction indicates a unit check condition, issuing this instruction transfers detailed information about the unusual condition.

2. If a read home address and record 0 count instruction is executed without a unit check, a subsequent read diagnostic sense instruction to the same drive transfers the home address skip displacement byte data in diagnostic sense bytes 22 and 23. Also, presented in byte 1, bit 6, of these 24 bytes is the condition of the WRITE INHIBIT bit of the drive.

3. If a test I/O instruction indicates a not-ready/unit check condition, but a subsequent read diagnostic sense instruction to the same drive returns 0's in the first two bytes and bits 0 through 4 of the third byte (thereby indicating no error), the drive has gone ready.

The attachment resets the unusual condition indication (sense bits) and diagnostic sense bits that provide information about the unusual condition after the last diagnostic sense byte has been stored in the disk drive data field.

Initial Conditions

DDCR—Not used for this instruction.

DDCR—Not used for this instruction.

DDDF—Need not be initialized (destination field for 24 bytes of sense information).

DDDR—Must contain the address of the leftmost byte of the DDDF.

Ending Conditions

DDCF—Unchanged.

DDCR—Unchanged.

DDDF—Contains 24 bytes of diagnostic sense information.

DDDR—Contains the address of the last DDCF byte operated on plus 1 (that is, the beginning DDDR address plus 24).

3340/3344 Data Module Attention Control Reset Operation

The 5415 accepts level 5 interrupt whenever interrupts are enabled and one of the following occurs; on Model 12, the 3340 attachment turns the op-end indicator on whenever it is enabled and one of the following occurs:

- Someone presses the ATTENTION key on a 3340/3344 drive.
- A 3340/3344 drive goes from not-ready to ready condition.
- The R/W or READ switch on the 3344 has been moved from one position to the other, or is set to READ and initial microprogram load is performed (includes SIO IPL command).

To release this interrupt or op-end indication:

- Issue a read extended functional sense command to determine which drive caused the interrupt, then
- Issue a data module attention control reset instruction, specifying the drive that caused the interrupt or op-end indication.

Note: If more than one drive has an active data module attention status, each such drive will initiate a level 5 interrupt or turn on the op-end indicator. In such cases, each interrupt or indication must be handled separately. The data module attention control reset instruction does not reset unit check, scan hit, or scan equal indications.

Initial Conditions

DDCF—Unused

DDCR—Unused

DDDF—Unused

DDDR—Unused

Ending Conditions

DDCF—Unchanged

DDCR—Unchanged

DDDF—Unchanged

DDDR—Unchanged

Program Notes

- The dropping of attachment busy signals ending status.
- Op-end status is not posted at the end of execution.

3340/3344 Read Extended Functional Sense Operation

Issuing the read extended functional sense instruction transfers 2 extended functional sense bytes from the attachment to the storage position addressed by the DDDR and to the next higher position. Only bits 0 through 3 of the leftmost byte have any meaning; bits 4 through 7 of the same byte and the bits in the rightmost byte are not used; they will always be binary 0's.

If any bit is on, the attachment has posted a DM attention indication for the associated drive:

Bit	Drive
0	1
1	2
2	3 (Model 15; bit is off for Model 12)[1]
3	4 (Model 15; bit is off for Model 12)[1]

The DM attention indication is posted by the attachment whenever a drive goes from not-ready to ready, or a drive has performed a recalibrate operation initiated by the drive ATTENTION switch.[1]

Issuing a read extended functional sense instruction does not reset any unit check or scan information. There will be no op-end interrupt pending or op-end indication posted.

Initial Conditions

DDCF—Not used.

DDCR—Not used.

DDDF—Need not be initialized; reserve 2-byte field.

DDDR—Must contain address of leftmost byte of 2-byte field to receive sense bytes.

[1]When 3344 disk drives are installed for drives 3 and 4, bits 2 and 3 have additional meaning. For 3344 drives, the DM attention is posted by moving the R/W or READ switch, or is set to READ when IMPL is performed (includes SIO IPL).

Ending Conditions

DDCF—Unchanged.

DDCR—Unchanged.

DDDF—Bits 0 through 3 of the byte stored in the addressed position contain sense information. All other bits are binary 0.

DDDR—Contains initial address plus 2.

Program Note

The attachment does not cause an op-end interrupt or turn on the op-end indicator at the end of this operation. When the attachment goes not busy the operation is complete.

3340/3344 Read and Reset Buffered Log Operation

The read and reset buffered log instruction transfers 24 bytes of data from usage counters in the attachment to the main storage DDDF, then resets the usage counters.

The data transferred is identical to the 3340/3344 sense information format 6 except that bytes 18 through 23 contain 0's (they are unused).

Initial Conditions

DDCF—Not used.

DDCR—Not used.

DDDF—Destination area for the 24 status bytes.

DDDR—Must contain address of leftmost byte in the DDDF.

Ending Conditions

DDCF—Unchanged.

DDCR—Unchanged.

DDDF—Contains 24 bytes of status information read from the attachment.

DDDR—Contains the address of the last DDDF location operated on plus 1, or initial DDDF address plus 24.

3340 Scan Read-OR Equal Operation (Models 12 and 15) and 3340 Scan Equal Operation (Model 12 only)

Before discussing the scan operation, it is necessary to define the use of the hex FF in the DDDF, and define *scan hit* and *scan field*:

FF — This code designates a noncompare position in the DDDF and can indicate either end of a scan field.

Scan Field — A scan field can be defined as:

- All the characters between the start of the DDDF and the first FF in the DDDF.
- All the characters between any FF and the next FF in the DDDF. (If multiple FF bytes appear in sequence in the DDDF, the FF bytes between the leading FF and trailing FF *do not* constitute a scan field.)

Scan Hit — A scan hit occurs whenever the scan field from the DDDF being compared with associated characters being read from the data area of the active record meets the conditions established by the instruction. In this case, a scan hit occurs when the scan field equals the data being read from the data area of the active record.

The scan read-OR equal instruction initiates a multiple-record operation. During the operation, the attachment compares a series of fields from the disk drive data field with associated fields being read from the data areas of successive sequential records. The scan portion ends as the result of one of the following conditions:

1. A scan hit occurs.
2. The last record specified by the disk control N-byte has been scanned without a scan hit occurring.
3. The attachment detects a check condition.

Whenever a scan hit occurs on a scan read OR equal operation the attachment reads the remaining data from the data field of the record being scanned, eventually storing it either 2 or 3 bytes after the first scan field in the original disk drive data field in main storage, and sets a scan hit indicator that can be tested by a TIO instruction.

To perform a scan read-OR equal operation on the Model 15, the attachment performs all the following functions. To perform a scan equal operation on the Model 12, the attachment performs items 1, 2, 3, and 7:

1. Reads the DDCF and DDDF into the attachment buffer from main storage.

2. On a byte-by-byte basis, compares characters in the first scan field in the buffered DDDF with associated characters being read from the data area of the record being read. (Note that the key field is never read by this instruction.)

3. If all the characters compare equal, performs the read portion of the operation, starting at step 4. If at least one character in the scan field was different from the associated character read from the data area on the record, the attachment operates as follows:
   a. If the field examined was the last scan field in the DDDF, and the record was the last record in the operation, the attachment immediately performs step 5.
   b. If the field examined was the last scan field in the DDDF, and there are more records to be scanned, the attachment updates the DDCF in the buffer, locates the next record, and starts scanning that record (return to step 2 of the scan operation).
   c. If the field examined was not the last scan field in the DDDF, the attachment compares the characters on the next scan field with associated characters being read from the data area of the active record, then repeats this step (step 3) of the operation.

4. Reads the remaining bytes from the data field on the active record into the buffered DDDF. The attachment starts reading at the byte associated with the first FF code, the code defining the end of the successful scan field. (Note that in this case the attachment *does not* bypass this byte from the data area on the record.) The attachment places this data in the leftmost bytes of the buffered DDDF, overlaying part of the original contents.

5. The attachment reads the contents of the buffered DDCF back into the DDCF in main storage. The DDCF will indicate the last record operated on.

6. Reads the contents of the buffered DDDF back into the DDDF in main storage, overlaying the main storage DDCF data from X upward in storage, where

   X = the first even-numbered position to the right of the first FF code in the main storage DDDF. This preserves the first scan field in the DDDF.

7. Sets a scan-hit indication and ends the operation.

3340/3344 Unconditional Scan Read Operation

If the first byte of the DDDF contains hex FF, the attachment immediately posts a scan-hit indication and transfers the entire data area from the first record to the DDDF in main storage as described in step 6 of the operation. The operation ends after a single record has been operated on, regardless of the contents of the N-byte in the DDCF. This function provides a means of reading the data area of a single record without reading the key area.

Program Note

As with all multiple record operations, if the DDCF specifies R0 as the starting record, the attachment scans even R0 and ends the operation after that record has been scanned. At the end of the operation the DDCF N-byte in main storage contains hex FF, regardless of its original content.

Example 1. Scan Hit Occurs on First Scan Field of Third Record Scanned

Contents of Main Storage DDDF at Start of Operation:

0	1	4	3	FF	J	G	J	O	N	E	S	FF

05F0

FF	FF	2	1	3	ƀ	ƀ	F	I	R	S	T

Contents of Attachment DDDF at Start of Operation:

0	1	4	3	FF	J	G	J	O	N	E	S	FF

FF	FF	2	1	3	ƀ	ƀ	F	I	R	S	T

Contents of Data Area in Third Record Scanned by Operation:

0	1	4	3	ƀ	1	8	.	0	0	ƀ	0	2	G

R	#	5	7	2	ƀ	S	O	C	K	E

Contents of Attachment DDDF Transferred to Main Storage DDDF:

ƀ	1	8	.	0	0	ƀ	0	2	G	R	#	5	7

2	ƀ	S	O	C	K	E	T	ƀ	W	R

Contents of Main Storage DDDF at End of Operation (first FF on even address):

0	1	4	3	FF	J	ƀ	1	8	.	0	0	ƀ	0

05F0 05F4

2	G	R	ƀ	S	O	C	K	E	T	ƀ

Contents of Main Storage DDDF at End of Operation (first FF on odd address):

0	1	4	3	FF	ƀ	1	8	.	0	0	ƀ	0	2

05F1 05F5

G	R	ƀ	S	O	C	K	E	T	ƀ

DDCF Initialized Values:

R-Byte = Record 6
N-Byte = 9

DDCF Residuals:

R-Byte = Record 8
N-Byte = 7

Example 2. Scan Hit Occurs on Second Scan Field

Contents of Main Storage DDDF at Start of Operation:

0	1	4	3	FF	J	G	J	O	N	E	S	FF

FF	FF	2	1	3	ƀ	ƀ	F	I	R	S	T

Contents of Attachment DDDF at Start of Operation:

0	1	4	3	FF	J	G	J	O	N	E	S	FF

FF	FF	2	1	3	ƀ	ƀ	F	I	R	S	T

Contents of Data Area of Record With Scan Hit:

0	8	2	4	ƀ	J	G	J	O	N	E	S	ƀ	ƀ

ƀ	ƀ	ƀ	1	4	8	#	4	1	8	6

Contents of Attachment DDDF Transferred to Main Storage DDDF:

ƀ	ƀ	ƀ	ƀ	ƀ	1	4	8	#	4	1	8	6	etc

Contents of Main Storage DDDF at End of Operation (first FF on even address):

0	1	4	3	FF	J	ƀ	ƀ	ƀ	ƀ	ƀ	1	4

05F0 05F4

8	#	4	1	8	6	etc

Contents of Main Storage DDDF at End of Operation (first FF on odd address):

0	1	4	3	FF	ƀ	ƀ	ƀ	ƀ	ƀ	1	4	8

05F1 05F5

#	4	1	8	6	etc

Initial Conditions

DDCF—Must identify the starting record and the number of additional records to be scanned.

DDCR—Must contain the address of the leftmost byte of the DDCF in main storage.

DDDF—Must be 258 (decimal) bytes long. Must contain scan arguments (data to be compared in each scan field) and FF bytes used to identify the end of each scan field or to identify positions in the record data area containing characters that are not to be compared. Positions not to be used to contain scan fields should be filled with hex FF. The last byte (that is, the rightmost byte) in the DDDF must contain FF.

DDDR—Must contain the address of the leftmost byte of the main storage DDDF.

Ending Conditions

DDCF—If a scan hit occurred during the operation, the DDCF contains the address of the record being processed when the scan hit occurred. If a scan hit did not occur, and no data check occurred, the DDCF contains the address of the last record scanned (initialized DDCF N-byte plus 1).

DDCR—Contains the address of the leftmost byte of the DDCF, as initialized.

DDDF—If a scan hit occurred, the DDDF contains the initial data up to and including the first hex FF and all the data residing in the attachment DDDF at the end of the read portion of the operation. The attachment reads its DDDF data into contiguous positions of the main storage DDCF, starting at the first FF code address plus 1 if the first FF is stored in an odd-numbered position, or at the first FF code plus 2 if the dirst FF is stored in an even-numbered position. All data stored in the initialized DDDF beyond the first FF will have been effectively shifted to the right by the number of characters in each scan field that did not cause a scan hit plus the number of FF bytes prior to the FF ending the scan field that resulted in a scan hit.

DDDR—Not changed.

3340/3344 Scan Read-OR High or Equal Operation (Models 12 and 15)

During a scan read-OR high or equal operation the attachment posts a scan hit whenever the hex value of a record data area either equals or exceeds the hex value of the corresponding scan field from the DDDF. Otherwise, the scan read-OR high or equal operation is identical to the scan read-OR equal operation.

The program can determine whether a scan hit was caused by a high condition or an equal condition by issuing a sense I/O instruction. If the field from the data area was higher than the scan field, the attachment will not have posted a scan equal indication (byte 1, bit 1 on). For an equal condition, this bit will be on.

3340 Scan High or Equal—Model 12 Only

The scan high or equal command functions are similar to the scan equal command. However, for scan high or equal, a scan hit occurs if the disk data field is higher than or equal to the storage data field. To determine whether the high condition or the equal condition caused a hit, issue a sense command testing the scan equal condition. If the scan equal bit is on, the compare is equal; if the bit is off, the compare is high.

3340/3344 Write Home Address and Record 0 Operation

System/3 has two write HA and R0 instructions: one for the HA and R0 on the even halftrack, and one for the HA and R0 on the odd halftrack. These instructions let the program modify the home addresses and the R0 count fields, and are usually used to flag tracks as defective or alternate.

Before issuing a write HA and R0 instruction, the system program must issue a read HA and R0 instruction for the same halftrack. If the read operation is performed without a data check, the write HA and R0 instruction can be issued immediately. If a data check occurs, the program can reissue the read instruction; if the data check persists, the cause could be a bad spot in the HA area or R0 count area of the halftrack. In this case, it may be possible to bypass the bad spot by following this procedure:

1. a. Move binary 100 into bits 0, 1, and 2, and 1 into bit 6 of the DDCF flag byte. (Bit 6 = 1 indicates the track is defective.)
   b. Issue a write HA and R0 instruction for the appropriate halftrack.
   c. Issue a read HA and R0 instruction for the same halftrack. If no data check occurs during this read operation, the write operation was successful and the procedure ends. If a data check occurs, perform step 2.

2. a. Move binary 010 into bits 0, 1, and 2, and 1 into bit 6 of the DDCF flag byte.
   b. Issue a write HA and R0 instruction for the appropriate halftrack.
   c. Issue a read HA and R0 instruction for the same halftrack. If no data check occurs during this read operation, the write operation was successful and the procedure ends. If a data check occurs, perform step 3.

3. a. Move binary 001 into bits 0, 1, and 2, and 1 into bit 6 of the DDCF flag byte.
   b. Issue a write HA and R0 instruction for the appropriate halftrack.
   c. Issue a read HA and R0 instruction for the same halftrack. If no data check occurs during this read operation, the write operation was successful and the procedure ends. If a data check occurs, the track is faulty and *should never be accessed.*

Throughout the procedure, the program must retain the original key length and data length values; otherwise, the attachment will not be able to bypass any bad spots in the HA and R0 areas of the halftrack.

During execution of the write HA and R0 instruction, the controller locates the appropriate halftrack (by sensing and recognizing the even or odd index marker), then writes the HA and R0 count area from the DDCF and the R0 key and data areas from the DDDF. After writing the HA and the R0 and their associated gaps, the attachment fills the remainder of the selected halftrack with hex 00 bytes.

Program Notes

1. The data module is shipped with the HA and R0 already written on each halftrack. (The key length for this initialized R0 is hex 00, and the data length hex 0008.)

2. Before issuing a write HA and R0 instruction, the program must load the appropriate halftrack identification and R0 count field information into the DDCF.

3. If either halftrack is flagged defective during the operation, the system program should:
   a. Flag the other halftrack as defective.
   b. Assign an alternate track for the defective track.
   c. Load the alternate track address into the R0 even count area and the R0 odd count area, using write R0 count key data and write R0 odd instructions.
   d. Identify the alternate track as such in the HA and R0 of each halftrack on the alternate track.
   e. Write the address of the defective logical track in the ID field of both the odd R0 and the even R0 of the alternate track.
   f. Write appropriate key and data fields in both the even R0 and the odd R0 of the alternate track.

   Flag byte bits 0, 1, and 2 to be written on the disk are generated by the attachment; they are not copied from the DDCF.

4. Write HA and R0 instructions cannot initiate multiple-record operations. The attachment ignores the DDCF N-byte.

Initial Conditions

DDCF—Contains the FCCHHR KL DL DL N field specifications for HA and R0:

FCCHH—Flag byte and track ID for home address and record 0.

R—Not used for this instruction; can contain any value.

KL DL DL—Record 0 key length and data length specifications. The total number of bytes in the key length and the data length fields must not exceed 256 (the key length may be 0). A sum greater than 256 results in a command reject status and the operation ends.

N—Not used for this instruction; can contain any value.

DDCR—Must contain the address of the leftmost byte of the DDCF.

DDDF—Contains contiguous R0 key and data fields.

DDDR—Must contain address of the leftmost byte of the DDDF.

Ending Conditions

DDCF—Contains the original contents with N unchanged.

DDCR—Contains the initialized address.

DDDF—The original contents are unchanged.

DDDR—Contains the initialized address of the DDDF.

3340/3344 Write Count Key Data Operation

This is a single full track initialization operation used to format single or multiple fixed format records. The disk drive starts formatting records at the record specified by the record identifier in the DDCF and formats n+1 records. The drive formats the count, key, and data areas as specified by the DDCF. The FCCHHR bytes of the count area are obtained from the DDCF. Key and data fields to be written are obtained from contiguous positions within the DDDF. Corresponding field length counts, KL and DL, are obtained from the DDCF. Before starting to write, the attachment calculates KL plus DL. If the sum is greater than 256 (decimal), the attachment sets the command reject status bit and ends the operation.

To write any record (Rn) except record 0, the attachment:

1. Locates the even index marker, then spaces over records until record Rn–1 has been passed (where Rn = the first record to be formatted by the instruction). As the attachment spaces over records, it checks for the correct FFCCHHR data in the count fields: if the attachment does not detect the correct count area while reading the entire track, it posts a no-record-found indication and ends the operation.

2. Calculates, on a record-by-record basis, whether the next record can be written completely on the current halftrack. If not, the attachment pads the remainder of the halftrack with hex 00 bytes, then does one of the following:
   a. If the current halftrack is the even halftrack, the attachment reads the odd index marker, the odd HA and the odd R0 and checks to determine that the odd HA and R0 FCCHH bytes match the FCCHH bytes from the last count field on the even halftrack. If they are not equal, the attachment posts an invalid track format indication. If they are equal, the operation continues.
   b. If the current halftrack is the odd halftrack, the attachment posts an invalid-track-format indication (bit 1, byte 1 returned to a diagnostic sense instruction) and ends the operation.

3. Writes record Rn, then subtracts 1 from the N-byte in the disk drive control field.

4. Checks the disk drive control field N-byte:
   a. If the N-byte contains FF, the attachment writes hex 00 bytes from the last record written to the end of the current halftrack. *Note that hex 00 is not written to the end of the entire track if the last record written was located in the even halftrack.* Therefore, if data was written onto the odd halftrack during prior operations, and if this data is to be removed, the program should issue a write count key data instruction specifying the first record after R0 odd as the record number (R) in the disk drive control field, or should issue a write HA and R0 odd instruction for that halftrack.
   b. If the N-byte contains a number other than FF, the attachment adds 1 to the value in the record number byte (R) in the disk drive control field, then repeats steps 3, 4, and 5 of this procedure until all records specified have been written.

Whenever the DDCF R-byte specifies hex 00 when the program issues a write count key data instruction, the attachment performs a single-record operation on R0 as follows:

1. Locates the sync byte and bits 6 and 7 of the even HA flag byte. (The CCHH bytes in the HA and in the disk drive control field need not be equal.)

2. Writes even R0; then sets the N-byte in the attachment disk drive control field to hex FF.

3. Writes hex 00 bytes from even R0 through the last byte on the even halftrack. *Note that this operation does not fill the bytes on the odd halftrack with hex 0's.* Therefore, if data was written onto the odd halftrack during prior operations, and if this data is to be removed, the program should issue a write count key data instruction specifying the first record after odd R0 as the record number (R) in the disk drive control field, or should issue a write HA and R0 odd instruction for that halftrack. Track initialization can be verified by issuing a read verify command.

Initial Conditions

DDCF—Contains the initial control field bytes (FCCHHR KL DL DL N) used to specify the starting record address, key and data length counts and the number of records (N+1) to be written.

DDCR—Must contain the address of the leftmost byte of the DDCF.

DDDF—Contains the information for contiguous key and data fields of the record to be written. If a multiple record operation is specified (original N-byte content was other than hex 00) the content of this field is written into the key and/or data areas of each record written by the operation. Hence, all records written have identical key areas and identical data areas.

DDDR—Must contain the leftmost byte address of the DDDF.

Ending Conditions

DDCF—Identifier portion contains address of last record written, N-byte contains FF.

DDCR—Contains the initialized DDCF address.

DDDF—Contents remain unchanged.

DDDR—Contains the initialized DDDF address.

3340/3344 Write Key Data Operation

The write key data operation transfers specified key and data fields from main storage to the selected disk drive and track. The attachment compares the flag and identifier field (FCCHHR) of the DDCF with the same flag, and identifier field of the count area read from the selected track. Comparison begins with the first count area read. A successful comparison is called record orientation. Following record orientation, the attachment compares the KL DL DL bytes from the disk drive control field with the KL DL DL bytes from the disk record count field. If the bytes are equal, the attachment writes the number of key and data bytes specified into the key and data fields of the oriented record, using data from the disk drive data field. If the KL DL DL bytes do not compare equal, the attachment sets the no record found status bit and ends the operation.

As the drive writes each record, it generates check field bytes and appends them to each key or data field, as required.

The drive writes multiple fixed format consecutive records if the DDCF N-byte is greater than 0. (Whenever the N-byte specifies the operation ends after one record has been written.) When a multiple-record operation is specified, the attachment updates its DDCF by adding 1 to the record number (R-byte) and subtracting 1 from the N-byte as each record is operated on.

The drive bypasses the R0 beyond any index it passes during multiple record operations. That is, if the odd index marker passes the read head after the operation has started reading records from the even halftrack, but before all the records have been read, the attachment ignores the odd R0 on the track and starts reading again when the next record is at the read/write head. Also, if the operation overflows from one track to the next, the attachment ignores the even R0 the new track, starting writing again at R1 after head switching has occurred.

Initial Conditions

DDCF—Contains the initial control field bytes (FCCHHR KL DL DL N). Specifies the starting record address, key and data length counts, and the number of records (N+1) to be written.

DDCR—Must contain the address of the leftmost byte of the DDCF.

DDDF—Contains contiguous key and data fields to be written into disk storage. Length = (N+1) (KL + DL)

DDDR—Must contain the address of the leftmost byte of the DDDF.

Ending Conditions

DDCF—Contains the address of the last record written or attempted to be written.

DDCR—Contains the initialized leftmost byte address of the DDCF.

DDDF—Contents remain unchanged.

DDDR—Contains the address of the last DDDF position operated on plus 1, or initialized value + (N+1) x (KL + DL)

3340/3344 Write Repeat Key Data Operation

The write repeat key data instruction writes data from the same main storage bytes into every record specified by the instruction. (At the start of the operation, the attachment stores the contents of the disk drive data field in a buffer, using this buffered data to write the key and data areas on every record handled by the instruction.)

If record 0 is specified as the starting record (hex 00 resides in the disk drive control field R-byte when the write repeat key data instruction is issued), only record 0 is written, regardless of the original content of the DDCF N-byte.

Initial Conditions

DDCF—Contains the initial control field bytes FCCHHR KL DL DL N. Specifies the starting record address, key and data length counts, and the number of records (N+1) to be written.

DDCR—Must contain the address of the leftmost byte of the DDCF.

DDDF—Contains contiguous key and data fields to be written into disk storage. Note that in this operation the disk drive data field contains data for one key area and one data area, and that these two fields are duplicated into the key and data areas of every record. Length = (KL + DL) x 1

DDDR—Must contain the address of the leftmost byte of the DDDF.

Ending Conditions

DDCF—Contains the address of the last record written or attempted to be written. For any condition except attachment check, the N-byte contains hex FF or the number of records still to be processed.

DDCR—Contains the initialized address of the DDCF.

DDDF—Contents remain unchanged.

DDDR—Contains the address of the last DDDF position plus 1, or initialized value + KL + DL.

3340/3344 Write Record 0 Odd Operation

The write record 0 odd instruction writes the odd record 0 count, key, and data areas on the track to which a seek has been performed. At the start of the operation the attachment stores the contents of the DDCF in its buffer, then locates the odd R0 by recognizing the odd index marker and bypassing the odd HA and its trailing gap. After locating odd R0, the attachment writes the count field from the DDCF and the key and data fields from the DDDF into the odd record zero count, key, and data areas. (If the count field specifies a KL byte of hex 00, no key area will be written in the record.) After writing odd record 0 and its trailing gap, the attachment fills all following bytes on the odd halftrack with hex 00. This erases any data previously written in those bytes.

The attachment does not use the R-byte or the N-byte for this operation, so these bytes can contain any values. At the end of the operation, the attachment returns the buffered DDCF value to the DDCF in main storage.

This instruction is usually used during initialization procedures to format the disk and, later, to assign alternate tracks.

Initial Conditions

DDCF—Contains the data to be stored in the odd record 0 count field:

F—Identifies the track being written as good primary, bad primary, good alternate, or bad alternate track. Bit 5 is not used for this instruction.

CCHH—Depends on track condition and type:
Good primary: Contain this logical track address.
Bad primary: Contain the address of alternate track to be sought.
Good alternate: Contain the address of track being replaced.
Bad alternate: Bytes should have no meaning, because this track should never again be addressed. Bytes can contain address of alternate track to be sought.

R—Not used; can contain any value.

KL—Contains the length of the odd record 0 key field. Must be hex 00 if IBM System/3 programming support is used.

DL DL—Contains the length of the odd record 0 data field. Must be hex 08 if IBM System/3 programming support is used.

N—Not used; can contain any value.

DDCR—Must contain the address of the leftmost byte of the DDCF.

DDDF—Contains contiguous key and data fields to be written into the odd R0.

DDDR—Must contain the address of the leftmost byte of the DDDF.

Ending Conditions

DDCF—Contents remain unchanged.

DDCR—Contains initialized address of the DDCF.

DDDF—Contents remain unchanged.

DDDR—Contains initialized address of the DDDF.

3340/3344 Write Count Compressed Data Operation

This is a multiple-record operation used to format any multiple of 4 records on a track in compressed data format. Formatting starts at the record specified by the R-byte in the disk drive control field; this byte must contain the hex equivalent of decimal 1, 5, 9, 13, 17, 21, 25, 29, 33, 37, 41, or 45. The formatting operation continues until the number of records specified by the DDCF N-byte have been formatted; the N-byte value plus 1 must be evenly divisible by 4, and head switching must not occur. This limits the number of records that can be formatted by a single instruction to 48, which is the track maximum (24 records per halftrack).

During the operation, the attachment:

1. Moves the DDCF data from main storage into attachment (control) storage.

2. Locates the even index, then spaces over records until the data field for record Rn−1 has been passed (where Rn = the first record to be formatted by the instruction). As the attachment spaces over records, it checks for the correct FCCHHR data in the count fields; if the attachment does not detect the correct count area while reading the entire track, it posts a no-record-found indication and ends the operation.

3. After spacing over the Rn−1 data area, the attachment starts writing compressed data record groups, starting with record Rn; the attachment uses the count field from the DDCF and the data field from the 256 (decimal) byte data field in the DDDF for the count and data areas of the records.
   a. After writing the count area for each record group, the attachment writes identical data into the four associated data areas. During the operation, the attachment updates the DDCF R-byte as required, so that after the fourth data area from each record group has been written on track, the R-byte contains the number of the next record to be written. The attachment concurrently subtracts 1 from the DDCF N-byte for each data area written so that the N-byte always contains the number of data areas (and therefore compressed format records) still to be written.
   b. If the operation does not end before the even index passes the read/write head, the attachment reads the odd HA and odd R0 and checks to determine that their FCCHH bytes match the FCCHH bytes from the last count field on the even halftrack. If they are not equal, the attachment posts an invalid track format indication; if they are equal, the operation continues. (This comparison occurs even though the R-byte in the instruction specifies a record in the odd halftrack.)
   c. When the N-byte contains hex FF after a record has been written, the attachment stops writing records and fills all remaining bytes in the current halftrack (note that only the current halftrack is filled) with hex 00.

4. The operation ends when the attachment encounters the next index marker and the attachment moves the data currently residing in its DDCF into the DDCF in main storage.

Program Notes

1. If record Rn−1 is not written in compressed format, the attachment posts invalid track format. *Exception:* When R1 is specified as the starting record, because R0 is never written in compressed format.

2. Track initialization can be verified by issuing a read verify command.

3. If the entire track is written in compressed format, the first (even) halftrack can then be written in standard format without disturbing the compressed data on the odd halftrack.

Initial Conditions

DDCF—Contains the initial control field bytes used to specify the starting record address, key and data lengths, and the number of additional records to be written:

F—Bit 5 must be 1.

CCHH—Must specify the logical track address.

R—Must specify the first record to be formatted; this must be decimal 1, 5, 9, 13, 17, 21, 25, 29, 33, 37, 41, or 45.

KL—Must be 00.

DL DL—Must be hex 0100.

N—Must indicate the number of records to be formatted minus 1; the total number of records to be formatted (N+1) must be evenly divisible by 4.

DDCR—Must contain the address of the leftmost byte of the DDCF.

DDDF—Contains 256 bytes of data to be written into each record formatted by the operation.

DDDR—Must specify the leftmost byte of the DDDF.

Ending Conditions

DDCF—CCHHR identifies the last record written, or attempted. N-byte will be FF if the operation ended successfully; otherwise, the N-byte value plus 1 indicates the number of records that were not successfully processed.

DDCR—Contains the initialized DDCF address.

DDDF—Unchanged.

DDDR—Contains the initialized DDDF address.

3340/3344 Programmed Attachment IPL

Issuing an SIO instruction that specifies IPL after an attachment check has occurred resets the controller and attachment, then reinitializes the attachment. This readies the attachment for subsequent operation.

Before issuing the SIO IPL instruction, the program should load the address of the leftmost byte of a 1,280-byte area of main storage into the DDDR. (This area is used by the attachment during the SIO IPL operation as a buffer to temporarily hold data from cylinder 00, head 00, records 25 through 29 or records 33 through 37 during the operation.) This 1,280-byte area must start on an even-byte boundary. Data in the area at the start of the operation will be destroyed, and the area can be used for other purposes after the operation.

At the start of the SIO IPL operation, the attachment sets attachment busy, resetting it at the end of the operation. The System/3 program that issued the SIO IPL instruction should monitor attachment busy. When the attachment goes not-busy, the program should branch to DDDR-I plus 4, where DDDR-I is the initial value set into the DDDR. If attachment busy is not reset within 1 second, the operation has failed and the program must execute a diagnostic LIO-1 instruction with EB1 (extended byte 1) bits 4 through 7 equal to hex 2 and EB2 bits 0 through 7 equal to hex 88, thereby stopping the attachment clocks. The program should then reissue the SIO IPL instruction.

After the attachment has been successfully reinitialized, the attachment returns control to the program at the address in the address recall register.

Note: If the DDDR contains the address of an odd-numbered storage position when SIO IPL is executed, a processor check (storage data bus in check) may occur.

Initial Conditions

DDCF—Unused.

DDCR—Unused.

DDDF—1,280-byte field that starts on an even-byte boundary. This field must not contain data that must be retained, for the data will be overlaid with micro instructions during the operation.

DDDR—Must contain the address of the leftmost byte of the 1,280-byte field used by the attachment as a buffer.

Ending Conditions

DDCF—Unchanged.

DDCR—Unchanged.

DDDF—Contains micro instructions that have been loaded into the microprocessor control storage.

DDDR—Contains initialized value plus 1,279 decimal (4FE hex).

3340/3344 Attachment and Drive Status Retrieval

Adapter and drive status are available to the program in the following ways.

Sense I/O Adapter Status

This instruction can be used to retrieve 2 bytes of adapter status information. These bytes contain all the data needed for usual attachment supervision (Figure 7-36). Additional status is needed only if a check condition occurs.

Sense I/O Diagnostic Status

This instruction provides in-depth diagnostic information about the adapter; it is used for CE diagnostic programming.

Read Diagnostic Sense Status

The read diagnostic sense status SIO instruction returns the status and condition of the drive in 24 bytes and seven different formats. Four formats—1, 4, 5, and 6—(Figures 7-26, 7-28, 7-29, 7-32, and 7-33) describe the 3340/3344; the remaining three formats (Figures 7-27 and 7-30) are associated with the attachment.

In each format, the first 8 bytes (0 through 7) provide information about status and condition. Sense byte 7 identifies the format in which the remaining bytes (8 through 23) are arrayed:

Byte 7

Bits 0-3 = The format array of bytes 8-24

0	1	2	3	Format
0	0	0	1	1
0	1	0	0	4
0	1	0	1	5
0	1	1	0	6

Bits 4-7 = Define a message

4	5	6	7	Message
0	0	0	0	0
↓	↓	↓	↓	↓
1	1	1	1	F

Byte \ Bit	0	1	2	3	4	5	6	7
0	Command reject	Intervention required	Not used	Equipment check	Data check	Overrun	Track condition check	Seek check
1	Permanent error[1]	Invalid track format	End of cylinder	Not used	No record found	Not used	Write inhibited	Operation incomplete
2	Not used	Correctable	Not used	Environmental data present	Not used		3344 Drive[2] / 70 M-byte DM[2]	
3	R-byte of last SIO to addressed drive[3]							
4	Q-byte of last SIO to addressed drive A	B	C	D	E	F	G	H
5	Low-order logical cylinder address[4] 128	64	32	16	8	4	2	1
6	High-order logical cylinder address	512	256	and	8	Logical track[4] 4	2	1
7	Format (bits 0-3 hex)				Message code (bits 4-7 hex)			

[1]Set by error recovery procedures (not used by attachment).

[2]If byte 2, bits 5, 6, and 7 are all off, the mounted DM is the CE module. If bits 6 and 7 are on, 3344 is installed.

[3]When the 3344 is installed and diagnostic status is presented for either drive 3 or 4, the R-byte has the following bit definitions:

```
Bits  0 1  2 3 4 5 6 7
      ---  -----------
            └Same as defined for 3340 R-byte
      0 0 — Error occurred in logical volume 1
      0 1 — Error occurred in logical volume 2
      1 0 — Error occurred in logical volume 3
      1 1 — Error occurred in logical volume 4
```

[4]These bytes indicate the residual address at the end of the operation. If byte 0, bit 7 (seek check bit) is on, these bytes indicate the current seek address. The current seek address is the last argument (address) issued to the device.

Figure 7-26. 3340/3344 Read Diagnostic Sense Bytes 0 through 7 Summary

MESSAGES[1]

	Format 0	Format 2[2]	Format 3
0	Error is defined in bytes 0-6.	Hardware-detected error caused interrupt.	[3]
1	Instruction issued had invalid Q- and R-byte combination.	End-of-trap count is off and end-of-file transfer is on.	Not used
2	Program issued write HA and R0 instruction before issuing read HA and R0 instruction to the same halftrack.	End-of-file transfer is off at end of field.	Not used
3	The drive addressed has a DM attention condition outstanding.	End-of-file transfer not on when expected during error checking and correction.	Not used
4	Adapter has detected an incorrect DDCF specification.	Channel counter check occurred during an ending procedure.	Not used
5	Not used	End-of-channel transfer not on when expected during ending procedure.	Not used
6	Not used	Difference counter equal 0 not on when expected during ending procedure.	Not used
7	Not used	Seek busy latch: —Did not set on seek or recalibrate —Did not reset after seek	Not used
8	Not used	Microprogram detected faulty scan compare detection hardware.	Not used
9	Not used	Microprocessor could not perform a successful readback check after an arithmetic operation on its DDCR or DDDR.	Not used
A	Not used	Not used	Not used
B	Command overrun occurred because adapter did not issue a command to controller soon enough.	Not used	Not used
C	Channel data overrun occurred.	Not used	Not used
D	Drive detected attempt to read, write, or scan on record other than R0 on a defective track.	Not used	Not used
E	Head switching occurred from an alternate track.	Not used	Not used
F	Not used	Not used	Not used

[1] Determined by format and message code (byte 7).
[2] Format 2 is used for adapter diagnostic programming by the CE.
[3] The attachment is unable to provide any sense data (bytes 0 through 23). System/3 programs generate Format 3, Message 0.

Figure 7-27. 3340/3344 Read Diagnostic Sense Byte 7 Formats 0, 2, and 3 Message Summary

MESSAGES[1] determined by format and message code (byte 7)

	Format 1	Format 4	Format 5
0	Unexpected drive status during operation	HA area data check	Not used
1	Transmit target error	Count area data check	Not used
2	Microprogram detected error	Key area data check	Not used
3	Transmit fixed head error or transmit difference high error (3344)	Data area uncorrectable data check	Data area correctable data check
4	Sync out timing error	HA area—no sync byte found	Not used
5	Unexpected drive status at initial selection	Count area—no sync byte found	Not used
6	Transmit cylinder address error	Key area—no sync byte found	Not used
7	Transmit head error	Data area—no sync byte found	Not used
8	Transmit difference error	Not used	Not used
9	Drive status not as expected during read IPL	Not used	Not used
A	Seek verification check on physical address	Not used	Not used
B	Seek incomplete	Not used	Not used
C	No interrupt from drive	Not used	Not used
D	Not used	Not used	Not used
E	DM incompatibility, invalid DM size	Not used	Not used
F	Not used	Not used	Not used

[1]Determined by format and message code (byte 7).

Figure 7-28. 3340/3344 Read Diagnostic Sense Byte 7 Formats 1, 4, and 5 Message Summary

	Byte \ Bit	0	1	2	3	4	5	6	7
Drive Status	8 (note 1)	Controller check	Device interface check	Drive check	Read/write check	On-line	Data module attention	Busy	Seek complete
Checks, Status	9	Data module loaded switch latched	Not used	Motor-at-speed latched	Air/belt switch latched	Write enabled	Fixed head 70 M-byte DM[2]	70 M-byte DM[2]	35 M-byte DM[1]
DM Sequence Control	10	Data module size check	Data module latch 4	Data module latch 2	Data module latch 1	Check latch	Data module sequence check latched	Bias disable switch	Odd track
Load Switch Status	11	Drive start switch	Data module present switch	Cover locked switch	Data module unloaded switch	Data module loaded switch	Air/belt switch	Carriage home	Motor-at-speed switch
R/W Safety	12	Multiple head select check	Capable/enable check	Write overrun	Index check	R/W interlock check	Control check	Transition check	Write current check
	13 (note 2)		Control interface bus out (for message codes 2[1] and C)				Expected drive status/data (for message codes 1, 3, 5, 6, 7, 8, and 9)		
	14 (note 2)		Control interface bus in (at the time an error was detected)						
	15		Control interface tag bus (at the time an error was detected)						
Access Status	16	Access timeout check	Overshoot check	Servo off-track check	Track crossing	Servo latch	Linear mode latch	Control latch	Wait latch
Controller Checks	17	PLO check	No PLO input	SERDES check	Gap counter check	Write data check	Monitor check	ECC check	ECC zeros detected
Micro Detected Errors	18 (note 3)					Coded error condition (bits 4-7 hex)			
Status	19	Set R/W on				Lo gain error		Fixed head feature	
Interface Checks	20 (note 4)	Control interface tag bus parity check	Control interface bus out parity check	Drive selection check	Device bus in parity check	Control interface bus in parity check	Write fail	Device bus out parity check	Device tag parity check
	21								
	22		Fault symptom code						
	23		Fault symptom code						

[1]Bytes 13, 14, and 15 are valid for messages 1, 3, 5, and 6 shown for microprogram error messages, determined by sense byte 18, bits 4-7.
[2]When bits 5, 6, and 7 are all 0, the DM indicated is the CE module.

Figure 7-29 (Part 1 of 3). Read Diagnostic Sense Bytes 8-23 Format 1 Summary (3340 only)

	Byte \ Bit	0	1	2	3	4	5	6	7
Drive Status	8 (note 1)	Controller check	Device interface check	Drive check	Read/write check	On-line	Spindle attention	Busy	Seek complete/ search sector
Checks, Status	9		Sector compare check	Motor-at-speed latched	Air/belt switch latched	Write enable	Fixed head installed	Always on	Always on
Sequence Control	10		Spindle sequence latch 4	Spindle sequence latch 2	Spindle sequence latch 1	Timer check latch	Sequence check latched		Odd physical track
Load Sw Status	11	Drive start latch	Guard band pattern	Target velocity	Track crossing		Air/belt switch		Motor-at-speed switch
R/W Safety	12	Multiple head select check	Capable/ enable check	Write overrun	Index check	Delta IW check	Control check	Transition check	Write current check
	13 (note 2)		Control interface bus out or Expected drive status/data (For message code C; if message code 2, see microprogram messages, bit 18. Valid for 1, 3, 5, 6, 7, 8, and 9)						
	14 (note 2)		Control interface bus in (at time error was detected) (Valid only for message codes 1, 3, 5, 6, 7, 8, and 9)						
	15		Control interface tag bus (at time error was detected) (Valid only for message codes 1, 3, 5, 6, 7, 8, and 9)						
Access Status	16	Access timeout check	Overshoot check	Servo off-track check	Rezero mode latch	Servo latch	Linear mode latch	Control latch	Wait latch
Controller Checks	17	PLO check	No PLO input	SERDES check	Gap counter check	Write data check	Monitor check	ECC check	ECC zeros detected
Micro Detected Errors	18 (note 3)						Coded error condition (bits 4-7 hex)		
Status	19	Set read/ write on (see byte 1)				Head short check			Fixed head feature
Interface Checks	20 (note 4)	Control interface tag bus parity check	Control interface bus out parity check	Drive selection check	Device bus in parity check	Control interface bus in parity check	Initialize write failure	Device bus out parity check	Device tag parity check
	21								
	22			Fault symptom code					
	23			Fault symptom code					

Figure 7-29 (Part 2 of 3). Read Diagnostic Sense Bytes 8-23 Format 1 Summary (3344 only)

Notes:

If busy is on (byte 8, bit 6), search track is in progress.

1. If set R/W is active (byte 19, bit 0), bits 5, 6, and 7 are as follows:

5	6	7
I write, sense	Index mark	Active track

2. If seek check is active (byte 0, bit 7), bytes 13 and 14 are as follows:

	0	1	2	3	4	5	6	7
Previous seek address Low logical cylinder address	128	64	32	16	8	4	2	1
Previous seek address High logical cylinder address (bits 0–3); Logical track address (bits 4–7)		512	256		8	4	2	1

Previous seek address is the address at which the drive was located prior to the last issued seek argument (bytes 5 and 6).

3. Byte 18, bits 4 through 7 specify one of the following coded error condition messages

0	Not used
1	No tag valid on R/W op
2	No normal or check end on R/W op or on ECC op
3	No response from controller on control op
4	Timeout waiting for index or active track
5	ECC hardware check
6	Multiple or no controllers selected
7	Preselection check
8	Head switch timer expired check
9	Busy missing after seek start is issued
A	Physical address
B-E	Not used
F	Attention check

4. If seek verification check is active on 3340 (byte 0, bit 7), bytes 20 and 21 are as follows:

	0	1	2	3	4	5	6	7
Present address (read from disk) Low logical cylinder address	128	64	32	16	8	4	2	1
Present address (read from disk) High logical cylinder address (bits 0–3); Logical track address (bits 4–7)		512	256		8	4	2	1

If seek verification check occurs on 3344 (format 1, message A), bytes 20 and 21 are:

	0	1	2	3	4	5	6	7
Present seek address (PA1 byte read from disk)	128	64	32	16	8	4	2	1
(PA2 byte read from disk)								

Figure 7-29 (Part 3 of 3). 3340/3344 Read Diagnostic Sense Bytes 8-23 Format 1 Summary

Byte \ Bit	0	1	2	3	4	5	6	7
8	Cycle steal overrun	Channel in parity check	Channel transfer check	Not used	Any external check	Not used	Not used	RCS reg. parity check
9	Sync out	Recycle	Timeout	File transfer check	FBO/FO reg parity check	FTO reg parity check	Not used	FBI reg parity check
10	Select active	Tag valid	Check end	CE alert	Normal end	Sync in	Index latch	Error alert
11	Attachment busy	Difference counter equal 0	End of channel transfer	Allow channel transfer	Seek complete drive 1	Seek complete drive 2	Seek complete drive 3	Seek complete drive 4
12	System or power on reset	Channel check reset	Force error mode	Not used	End of trap count	Scan hit	Scan equal	End of file count
13	File bus out register (at time of error)							
14	File bus in register (at time of error)							
15	File tag out register (at time of error)							
16	Scan command	1 = High or equal 0 = Equal	Scan split field	Last record	Allow file transfer	File odd transfer	1 = Data to file	Inhibit file to control store transfer
17	1 = Data to channel	Channel odd transfer	1 = LSR cycle steal request	1 = DDDR	Allow diff. cntr. channel	Allow diff. cntr. file	Subtract	Channel one byte transfer
18	Not used							
19	Not used							
20	Not used							
21	Not used							
22	Not used							
23	Not used							

Figure 7-30. 3340/3344 Read Diagnostic Sense Bytes 8-23 Format 2 Summary

Byte \ Bit	0	1	2	3	4	5	6	7
8[1]	Cylinder address							
9[1]	Cylinder address							
10[1]	Head address							
11[1]	Head address							
12[1]	Record number							
13	Sector number							
14								
15								
16								
17								
18								
19								
20								
21								
22	Fault symptom code							
23	Fault symptom code							

[1] Count Identification

Figure 7-31. 3340/3344 Read Diagnostic Sense Bytes 8-23 Format 4 Summary

Byte \ Bit	0	1	2	3	4	5	6	7
8[1]	Cylinder address							
9[1]	Cylinder address							
10[1]	Head address							
11[1]	Head address							
12[1]	Record number							
13	Sector number							
14								
15	Restart displacement							
16	Restart displacement							
17	Restart displacement							
18	Error displacement							
19	Error displacement							
20	Error pattern							
21	Error pattern							
22	Error pattern							
23								

[1]Count Identification

Figure 7-32. 3340/3344 Read Diagnostic Sense Bytes 8-23 Format 5 Summary

Byte \ Bit	0	1	2	3	4	5	6	7
8[1]	Number of bytes read or scanned (key and data areas only)							
9	Number of bytes read or scanned (key and data areas only)							
10	Number of bytes read or scanned (key and data areas only)							
11	Number of bytes read or scanned (key and data areas only)							
12								
13								
14								
15								
16	Number of physical seeks performed							
17	Number of physical seeks performed							
18								
19								
20								
21								
22								
23								

[1]Count Identification

Figure 7-33. 3340/3344 Read Diagnostic Sense Bytes 8-23 Format 6 Summary

3340/3344 ERROR DETECTION, LOGGING, AND RECOVERY

3340/3344 Visual Indications

The system console panel provides processor check and I/O attention indicators. Visual indication of status is available from indicators on the 3340/3344.

3340/3344 Program Error Detection

Error conditions in either the attachment or the 3340/3344 subsystem can be detected by using the test I/O and sense I/O instructions.

If the attachment sense information indicates a not-ready/unit check condition, the program can issue a start I/O diagnostic sense instruction to retrieve 24 bytes of 3340/3344 sense information. The 24 bytes contain information required by the porgram for error recovery and by the CE for maintenance.

Hardware-detected error conditions that indicate a possible failure in the attachment microprocessor are presented as an attachment check in the attachment sense information. When this occurs, additional diagnostic information can be retrieved by issuing diagnostic load I/O and diagnostic sense I/O instructions.

3340/3344 Error Recording

The program should initiate a message to the system operator whenever a permanent error condition occurs. (A permanent error is considered to be one that cannot be recovered without operator action.) The message should contain enough information to let the operator identify the failing unit and identify the error condition as one of the following:

- Attachment check
- Equipment check
- Data check (noncorrectable)
- Overrun
- Seek check
- Drive not ready or CE mode (intervention required)
- Wrong data module size (intervention required)
- Write command to drive set up for read only mode
- Track condition check
- Command reject
- Invalid track format
- No record found
- End of cylinder

Unless prevented by a permanent hardware failure, any error condition that results in a console message should also be recorded. (If the system is being maintained by IBM customer engineers, such conditions must be recorded in the format and location specified in the following paragraphs.)

3340/3344 Usage and Error Log Required for IBM CE Maintenance

Information recorded in the usage and error log provides an indication of current reliability and helps customer engineers isolate failures. Systems being maintained by IBM customer engineers must prepare a log as described in this section. The 3340/3344 usage and error log occupies a reserved area (logical cylinder 209, logical heads 1 through 4) on the logical volume from which the IPL was performed. Volume 1 is on disk 3, R1; volume 2 on disk 3, F2. Each of these four reserved tracks is used to record information relating to a single disk drive, with heads 1 through 4 corresponding to drives 1 through 4. If any such track becomes defective, it must be assigned an alternate.

Each error log track must be initialized in compressed data format to contain 48 records, and each record is then divided into four areas of 64 bytes. The first and last areas on the error logging track are reserved for special program use. The first area (area 1 on record 1) stores the address of the newest log entry; the last area (area 4 on record 48) always contains hex 00. The remaining 190 64-byte areas are used for recording 3340/3344 usage and error statistics, and are called log entry areas. Each log entry area is formatted as follows:

Bytes	Content of Field
00-05	Volume ID (six EBCDIC characters)
06-11	Read usage count
12-15	Seek usage count
16-39	Diagnostic sense data
40	Error recovery procedure (ERP) retry count
41-42	Month
43-44	Day
45-46	Year
47-48	Hour on Model 15; unused on Model 12
49-50	Minute on Model 15; unused on Model 12
51-52	Second on Model 15; unused on Model 12
53-63	Unused

Log entries are created (error condition or volume ID change) for each drive in use, starting with the second 64-byte area of each track and continuing sequentially through the next to the last 64-byte area of the track. If the logging track should overflow, the oldest entries are to be overlayed, as required. The last log entry created will be followed by at least one entry of 64 hex 00 bytes. (These hex 00 bytes are in addition to those in area 4 of record 48.)

Procedure for 3340/3344 End of Job or Volume ID Change

1. Read area 1 of the first record in the track to locate the last log entry area used to record data other than hex 00 bytes.

2. Examine the seventh byte in the diagnostic sense data field in the log entry area.
   a. If the byte contains hex 60, the last log entry is not an error entry, but contains the accumulated number of bytes read and the accumulated number of seeks performed since an error was logged. Perform this procedure:
      (1) Issue a read and reset buffered log instruction. The attachment will return 24 bytes of diagnostic sense data (shown in *Read Diagnostic Sense Bytes 0 through 7* and in *Read Diagnostic Sense Byte Format 6 Summary*).
      (2) Write this data into the diagnostic sense data field of the log entry area.
      (3) Add the contents of the read usage count field from the log entry area and read diagnostic sense bytes 8 through 11, and write the total into the read usage count field of the log entry area.
      (4) Add the contents of the seek usage count field from the log entry area and read diagnostic sense bytes 16 and 17, then write the total into the seek usage count field of the log entry area.
      (5) Write current time and data information into appropriate log entry fields.
   b. If the byte contains other than hex 60, the last log entry is an error log entry that contains error diagnostic data plus the number of bytes read and seeks performed between permanent errors. Perform the following procedure:
      (1) Issue a read and reset buffered log instruction. The attachment will return 24 bytes of diagnostic data.
      (2) Write this data in the diagnostic-sense-data data field of the log entry field that is filled with hex 00 bytes.
      (3) Write data from read diagnostic sense bytes 8 through 11 into the third through sixth bytes of the read usage count field on the log entry area.
      (4) Write data from read diagnostic sense bytes 16 and 17 into the third and fourth byte of the seek usage count field on the log entry area.
      (5) Write all remaining log entry fields.
      (6) Write 64 hex 00 bytes in the next log entry field.
      (7) Update pointer by storing the address of the log entry area just used in the first 64-byte field on the track.

Procedure for 3340/3344 Permanent Error

1. Locate the last log entry area used to record error or usage data by reading the pointer stored in the first 64 bytes of R1 on the track.

2. Examine the seventh byte in the diagnostic sense data field in the log entry area.
   a. If the byte contains hex 60, the last log entry is not an error entry, but contains the accumulated number of bytes read and the accumulated number of seeks performed since an error was logged. Perform this procedure:
      (1) Issue a read and reset buffered log instruction.
      (2) Add the contents of the read usage count field from the log entry area to the value from read diagnostic sense bytes 8 through 11, and write the total into the read usage count field of the log entry area.
      (3) Write current time and data information into appropriate log entry fields.

(4) Determine the cause of the error. If other than attachment check, issue a read diagnostic status instruction, and write the 24 bytes of status information into the diagnostic sense data field of the log entry area. If the error was caused by an attachment check, fill the bytes of the diagnostic sense data field with the following information, which becomes read diagnostic sense byte 7 format 3:

Bytes	Data to be Written into Bytes by Program
0	Functional sense byte EB-2 (unit check and seek complete bits)
1	Functional sense byte EB-1 (scan equal, removable disk, etc)
2	IOP check sense (EB-1 byte of sense I/O diagnostic status after loading link address counter with hex 3 using diagnostic LIO-2 instruction)
3	IOP idle sense (EB-1 byte of sense I/O diagnostic status after loading link address counter with hex 2 using diagnostic LIO-2 instruction.)
4	Q-byte from last SIO prior to the error
5-6	Unused
7	Constant hex value 30
8-23	Unused

(5) Write 64 hex 00 bytes in the next log entry area.

b. If the byte contains other than hex 60, the last log entry is an error entry that contains error diagnostic data, the number of bytes read since the last permanent error, and the number of seeks performed since the last permanent error. Performing this procedure:

(1) Issue a read and reset buffered log instruction.

(2) Write the data from read diagnostic sense bytes 8 through 11 into bytes 3 through 6 of the read usage count field on the log entry area.

(3) Write the data from read diagnostic sense bytes 16 and 17 into bytes 3 and 4 of the seek usage count field on the log entry area.

(4) Perform steps 2a(3), 2a(4), and 2a(5) of this procedure.

Log Entry Programming Note

Data associated with different data modules should never be intermixed in a single log entry. Therefore, whenever the volume ID changes, the program should log subsequent count data and error data into a new log entry area, and write 64 hex 00 bytes into the next sequential log entry area. In all other respects, treat the log entry procedure as described in the preceding text.

SUGGESTED 3340/3344 ERROR RECOVERY PROCEDURES

Suggested priority for examining disk status and recovery actions are listed in Figure 7-34.

Priority	Byte[1]	Bit	Condition	Action
1	FS 1	7	Adapter check	VI
2	DS 2	3	Environmental data present	V
3	DS 0	1	Intervention required	II
4	DS 2 DS 6	5 7	Wrong data module size	II
5	DS 1	6	Write instruction addressed to a drive with DM switch set at READ ONLY	II
6	DS 0	6	Track condition check	VIII
7	DS 0	7	Seek check	III
8	DS 0	3	Equipment check	VII
9	DS 0	5	Overrun	VII
10	DS 1	4	No record found	IX
11	DS 1	1	Invalid track format	IX
12	DS 0	4	Data check	IV
13	DS 0	0	Command reject	IX
14	DS 1	2	End of cylinder, or end of logical volume (3344)	IX

[1]FS = Functional sense byte; DS = diagnostic sense byte

Figure 7-34 (Part 1 of 2). 3340/3344 Status-Checking Priorities and Suggested Recovery Actions

Action	
I	When the program does not attempt to recover from any of the following conditions: Data check — Invalid track format Overrun — End of cylinder Equipment check — No record found Command reject — Adapter check Perform the following procedure: 1. Post error completion. 2. Exit.
II	1. Test for a not-ready/unit check (TIO instruction) before issuing the next SIO. This test is not required for Model 12. 2. If: a. The unit is not ready (signalled by sense byte 0, bit 1), or b. Intervention is required because of wrong data module size (signalled by DS byte 2, bit 5 or 7), or c. The program issued a write instruction to a write-inhibited drive (signalled by DS byte 1, bit 6), issue a console message indicating the status of the drive causing the intervention required state. Otherwise, perform step 3. On Model 12, the CPU I/O ATTENTION light turns on for conditions a, b, and c. 3. If DS byte 10, bits 1, 2, and 3 are not all off, perform Action IX. *Note:* If an intervention required state occurs between the time the TIO and the SIO were issued, the I/O ATTENTION light turns on.
III	1. Recalibrate the failing drive. 2. Perform Action VII.
IV	1. If the error is not correctable, perform Action VII; if it is, advance to step 2. 2. Subtract the values stored in bytes 18 and 19 of format 5, message 3, from the residual value in the DDDR at the end of the operation resulting in the unit check. (The result will be the address of the leftmost of two bytes. This byte contains an error, and the byte to its right can contain an error also.) 3. Examine bytes 20 and 21 of format 5, message 3, bit by bit. For each bit that is on, reverse the bit residing in the corresponding bit location in the two error bytes located by step 1. (That is, if the bit in the error byte is 0, set it to 1; if the bit is 1, set it to 0.) 4. Perform Action X.
V	1. Read and reset buffered log. 2. Updates usage and error log. 3. Perform Action VII.
VI	1. If the microcode has been loaded three times, perform Action VII. Otherwise, advance to step 2. 2. Reload microcode, the return to step 1.
VII	1. If retry number has been reached, go to Action IX. Otherwise, advance to step 2. 2. Reissue original command, then return to step 1. *Note:* The program should retry the original command at least eight times.
VIII	1. Read HA and R0 on defective track. 2. If on a defective primary track, seek to the assigned alternate. 3. If switching from an alternate, seek to the defective primary plus 1. 4. Continue the operation.
IX	1. Log appropriate console message. 2. Update the usage and error log with data from the original error status bytes. 3. Wait for operator response.
X	1. Update the usage and error log with status from first error. 2. Continue program execution.

Figure 7-34 (Part 2 of 2). 3340/3344 Status-Checking Priorities and Suggested Recovery Actions

3348 Data Module and 3344 Data Storage Initialization

Initialization at the Plant of Manufacture

All 3348 data modules and 3344 data storage devices are initialized at the plant of manufacture for S/370 use:

- Home address records (HA) and track descriptor records (R0) are written on all tracks.
- All tracks in S/370 cylinders 696 and 697 decimal are flagged as alternate tracks (flag byte bit 7 = 1).
- On data modules, alternate tracks are not assigned to any defective tracks, because a pack is not shipped if it contains a defective track. The 3344 data storage device can be shipped with defective tracks that are flagged defective with alternate tracks assigned.
- A skippable defect causes the skip displacement bytes in the corresponding home address to be written.
- Written skip displacement bytes indicate to the using attachment that a defect must be skipped during normal operation.

A data module initialized for use on a S/370 must be re-initialized for System/3 use.

New 3348 Data Module or 3344 Data Storage Initialization for System/3

A new data module or 3344 data storage is initialized for use on a System/3 as follows:

- A new alternate track area (System/3 cylinder addresses 167 and 168 for the 3348 data module and 187 and 188 for the 3344 data storage) is assigned for systems using IBM System/3 programming support.
- All tracks in the new alternate track area are flagged as alternate tracks (flag byte bit 7 = 1).
- A new primary track area (System/3 logical cylinder addresses 00 through 166 and 169 through 209 for the 3348 data module, and addresses 00 through 186 and 189 through 209 for the 3344 data storage) is assigned.
- All primary (customer data) areas are written with the write count compressed data format command.
- The area assigned at the plant to alternate S/370 tracks is assigned to the primary (customer data) area.

See the System/3 track initialization procedures for detailed explanations of this operation.

Used 3348 Data Module or 3344 Data Storage Initialization for System/3

The System/3 uses:

- Two S/370-3340/3344 logical tracks to create one System/3-3340/3344 primary track.
- Two S/370-3340/3344 alternate tracks to create one System/3-3340/3344 alternate track.

The initialization of a used data module or 3344 data storage is identical to that of a new data module except that defective tracks and their assigned alternate tracks must be reassigned. Therefore, when a data module from a S/370-3340/3344 with a defective track is first placed on a System/3-3340/3344:

- Half of a System/3-3340/3344 primary track is flagged as defective, and the other half of the same track is flagged as good.
- Half of a System/3-3340/3344 alternate track is assigned to the defective track, and the other half of the same track is unassigned.

Neither a half defective track nor a half alternate track is permitted on a System/3-3340/3344. Therefore, the System/3 initialization program:

- Flags an entire track as defective.
- Assigns an entire alternate track to one defective track.

See System/3 track initialization procedures for detailed explanations of this operation.

Detailed Track Initialization for System/3

The System/3 track initialization and write data verification test program must be used to initialize all packs used on a System/3 that is using IBM programming support.

The System/3 track initialization and surface analysis program initializes the:

- New alternate tracks
- Primary (or customer data) tracks

New Alternate 3340/3344 Track Initialization for System/3

The System/3 new alternate tracks are initialized as follows:

1. A good alternate track is flagged good (flag byte = hex 05).

2. A defective alternate track is flagged defective (flag byte = hex 07).

3. All good alternate tracks are written by using the write count compressed data format command, full track, with the:
   a. Flag byte = hex 05
   b. Count area CCHH = home address area CCHH

The System/3 alternate tracks are contained in cylinder 167, head 0, through cylinder 168, head 19 for the 3348 data module; cylinder 187, head 0 through 188, head 19 for the 3344 data storage.

Primary (or Customer Data) Track Initialization for System/3: The primary (or customer data) tracks are initialized from the following S/370 track areas:

- Primary (or customer data) track area

- Alternate track area

S/370 Primary Track to S/3 Primary Track (on 3340/3344) Initialization

The System/3 primary tracks are initialized from the S/370 primary tracks as follows:

1. Two S/370 good primary tracks are flagged as one System/3 good primary track (flag byte = hex 04) and are written by using the commands:
   a. Write HA and R0 even
   b. Write HA and R0 odd

2. Two S/370 defective primary tracks are flagged as one System/3 defective primary track (flag byte = hex 06) and are written by using the commands:
   a. Read HA and R0 even
   b. Write HA and R0 even
   c. Write count, key, data, R0, with the count area CCHH containing the address of an unassigned System/3 alternate track in cylinder 167 or 168 on the 3348-70 data module and cylinder 187 or 188 on the 3344 data storage.
   d. Read HA and R0 odd
   e. Write HA and R0 odd
   f. Write R0 odd, with the count area CCHH containing the address of the same unassigned System/3 alternate track used by the write count, key, data, R0 command

3. All good primary tracks are written using the write count compressed data format command, full track, with the:
   a. Flag byte = hex 04
   b. Count area CCHH = home address area CCHH

The System/3 primary track area initialized from the S/370 primary track area is contained on:

1. 3348-70 Data Module
   a. Cylinder 00, head 0, through cylinder 166, head 19
   b. Cylinder 169, head 0, through cylinder 208, head 15

2. 3344 Data Storage
   a. Cylinder 00, head 0 through cylinder 186, head 19
   b. Cylinder 189, head 0 through cylinder 209, head 19

S/370 Alternate Track to S/3 Alternate Track (on 3340/3344) Initialization

The System/3 primary tracks are initialized from the S/370 alternate tracks as follows:

1. Two S/370 good alternate tracks are flagged as one good System/3 primary track (flag byte = hex 04) and are written by using the commands:
   a. Read HA and R0 even
   b. Write HA and R0 even
   c. Read HA and R0 odd
   d. Write HA and R0 odd

2. Two S/370 defective alternate tracks are flagged as one System/3 defective primary track (flag byte = hex 06) and are written by using the commands:
   a. Read HA and R0 even
   b. Write HA and R0 even
   c. Write count, key, data, R0, with the count area CCHH containing the address of an unassigned System/3 alternate track in cylinder 167 or 168 on the 3348-70 data module and cylinder 187 or 188 on the 3344 data storage.
   d. Read HA and R0 odd
   e. Write HA and R0 odd

f. Write R0 odd, with the count area CCHH containing the address of the same unassigned System/3 alternate track used by the write count, key, data, R0 command

3. All good primary tracks are written by using the write count compressed data format command, full track, with the:
   a. Flag byte = hex 04
   b. Count area CCHH = home address area CCHH

The System/3 primary track area initialized from the S/370 alternate track area is contained on cylinder 208, head 16, through cylinder 209, head 7.

3340/3344 VOLUME TABLE OF CONTENTS (VTOC) FOR SYSTEM/3 USING IBM PROGRAMMING SUPPORT

Each System/3-3348-70 data module or 3344 data storage contains a:

- System/3 VTOC similar in format to that of the 2316 with a 1000-file VTOC
- S/370-compatible volume label and a S/370 VTOC, both of which are written during System/3 data module or 3344 data storage initialization

Both of the above are written during System/3 data module or 3344 data storage initialization.

PREVENTION OF DATA DESTRUCTION

The S/370 contains one DSCB-1 (data storage control block). This block indicates that the System/3-3348-70 data module or 3344 data storage is valid (has not reached its expiration date), and therefore prevents the destruction of data on a valid module or storage. This DSCB-1 is a format 1 DSCB.

S/370-3348 DATA MODULE OR 3344 DATA STORAGE IDENTIFICATION

When a S/370-3348-70 data module or 3344 data storage is placed on a System/3, the system determines that it is data from a S/370 by examining the:

Volume label
VTOC
Format of the other areas on the 3348-70 data module or 3344 data storage

3340/3344 ALTERNATE TRACK ASSIGNMENT

During customer use, the 3348-70 data module or 3344 data storage may get a void, a scratch, or a foreign particle embedded in the disk surface. Any one of these defects can cause read errors every time that particular area (track) is read. If the defect is large enough, the error correction circuits cannot correct the error. The track must then be flagged defective and assigned an alternate track. The customer uses a utility program for this purpose.

The utility program rewrites the defective track with a:

- Read HA and R0 count even command
- Write HA and R0 even command (flag byte = hex 06)
- Write count, key, data, R0, with the count area CCHH containing the address of an unassigned System/3 alternate track in cylinder 167 or 168 for the 3348 data module and 187 or 188 for the 3344 data storage.
- Read HA and R0 count odd command
- Write HA and R0 odd command (flag byte = hex 06)
- Write R0 odd, with the count area CCHH containing the address of the same unassigned System/3 alternate track used by the write count, key, data, R0 command.

The same utility program also writes the alternate track with a:

- Read HA and R0 count even command
- Write HA and R0 even command (flag byte = hex 05)
- Write count, key, data, R0, with the count area CCHH containing the address of the defective System/3 track
- Read HA and R0 count odd command
- Write HA and R0 odd command (flag byte = hex 05)
- Write R0 odd, with the count area CCHH containing the address of the defective System/3 track

3340/3344 START I/O (SIO)

Op Code (hex)	Q-Byte (binary)	R-Byte (binary)
Byte 1	Byte 2	Byte 3
F3	110x x xxx	xxxx xxxx

DA M N — Control Code

N-Code	Bits 0123 4567	Function Specified	
000	0000 0000	Seek	
	0000 0001	Recalibrate	
001	0000 0000	Read key data	
	0000 0001	Read home address and record 0 count even	
	0000 0010	Read count key data	
	0000 0011	Read verify key data	
	0000 0100	Read count key data diagnostic (CE diagnostic)	
	0000 0101	Read and reset buffered log	
	0000 0111	Read diagnostic sense	
	0000 1000	Read record 0 key data odd	
	0000 1001	Read home address and record 0 count odd	
	0000 1011	Read extended functional sense[1]	
	0000 1101	Read reset data module attention control	
010	0000 0000	Write key data	
	0000 0001	Write home address and record 0 even	
	0000 0010	Write count key data	
	0000 0011	Write repeat key data	
	0000 0110	Write record 0 odd	
	0000 1000	Write count compressed data	
	0000 1001	Write home address and record 0 odd	
011	0000 0000	Scan equal (Model 12 only)	
	0000 0010	Scan high or equal (Model 12 only)	
	0000 1100	Scan: read if equal	
	0000 1101	Scan: read if high or equal	
		Model 15	*Model 12*
100[2]	1xxx xx00	Enable interrupt for all drives	Enable op-end indicators, all drives
	x1xx xxx0	Reset seek 1 complete	Reset seek 1 complete
	xx1x xxx0	Reset seek 2 complete	Reset seek 2 complete
	xxx1 xxx0	Reset seek 3 complete	Not used
	xxxx 1xx0	Reset seek 4 complete	Not used
	xxxx x1x0	Reset op end interrupt for all drives	Reset op-end indicators for all drives
	0xxx xx10	Disable interrupt for all drives	Disable op-end indicators for all drives
	0xxx xx01	IPL (load initial program)	IPL (load initial program)

Any N-code not shown is invalid and causes:
- Program check if interrupt level 7 is enabled on Model 15
- Processor check if interrupt level 7 is not enabled on Model 15
- Processor check on Model 12

Any x shown in the control code can be 1 for multiple control instructions.
Any control code not shown may result in the attachment hanging up in a busy state.

DA = 1100 and M = 0 specifies 3340 drive 1.
DA = 1100 and M = 1 specifies 3340 drive 2.
DA = 1101 and M = 0 specifies 3340/3344 drive 3; invalid if drive 3 is not installed.
DA = 1101 and M = 1 specifies 3340/3344 drive 4; invalid if drive 4 is not installed.
(Note that Q-bits 0, 1, 2, = 110 specifies the 3340/3344, while Q-bits 3 and 4 specify the drive.)

F3 specifies a start I/O operation. F as the first hex character in the op code identifies a command-type instruction (that is, an instruction without operand addressing).

---

[1] Any installed drive can be specified and instruction will be executed.
[2] Q-byte bits 3 and 4 (drive specification bits) are ignored; attachment circuits are addressed. However, an installed drive must be addressed.

Operation, General

The drive specified by the DA- and M-codes performs the function specified by the N-code and control code. *Exception:* When the N-code = 100, the SIO commands address all installed drives although seek interrupts still apply to individual, specified drives.

Program Notes

- Issuing any start I/O except interrupt or op-end control, read diagnostic sense, read extended sense, or read data module control to a busy attachment causes the program to loop on the instruction until the attachment becomes not-busy. *Exception:* If the Dual Program Feature is enabled, the processing unit activates the inactive program level. If the instruction addresses a drive that is not installed, a program check or processor check occurs with an invalid Q-byte indicated.

- The attachment provisionally accepts a single start I/O specifying read, write, or scan for later execution whenever the addressed drive is executing a seek. If an error occurs during the seek, the attachment aborts the provisionally accepted SIO. At the end of the seek operation, the attachment then sets no-op status bit, the unit check bit, and either a seek check bit or attachment check bit (as appropriate), and if op-end functions are enabled requests an interrupt on Model 15 or sets op-end indication on Model 12.

- A seek instruction on one drive can be overlapped with seek instructions on all other drives. A read, write, or scan on one drive can be overlapped with a seek instruction on any other drive if the seek instruction is issued first. Overlapping does not occur if the seek is issued during a read, write, or scan operation on any drive.

- The start I/O instruction uses the contents of the disk drive (data) address register (DDDR) as the initial main storage address of all disk record data fields. It uses the contents of the disk drive control (address) register as the address of the disk drive control field (DDCR) in main storage.

- The attachment always accepts an SIO interrupt/op-end-indicator control instruction, regardless of the status of the file or control unit. Issuing this SIO *does not* reset the attachment status.

- An SIO issued to a not-ready drive on Model 15 results in a unit check. An SIO issued to a not-ready drive on Model 12 results in an I/O attention condition, and when this condition is corrected a unit check occurs.

Op-End Interrupts and Op-End Indications

The attachment presents an op-end interrupt request to the Model 15 processing unit and sets the op-end indicator on Model 12 (if they are enabled) at the end of the processing unit instruction during which one of the following conditions occurred on the selected drive:

- The drive completed a data transfer operation (either read, write, or scan).

- The drive finished a seek operation.

- A read, write, or scan SIO was aborted because of an equipment check.

- An attachment check is pending.

Note: The attachment does not post an op-end interrupt or turn the op-end indicator on at the end of either a read extended functional sense operation or a data module attention control reset operation.

3340/3344 LOAD I/O (LIO)

Op Code (hex)	Q-Byte (binary)	Operand Address	
Byte 1	Byte 2	Byte 3	Byte 4
31	110x x xxx	Operand 1 address	
71	110x x xxx	Op 1 disp from XR1	
B1	110x x xxx	Op 1 disp from XR2	

DA M N

N-Code	To Be Loaded
100	Disk drive data register (DDDR)
101	CE diagnostic LIO 1
110	Disk drive control register (DDCR)
111	CE diagnostic LIO 2

Any N-code not shown is invalid and causes:
- Program check if interrupt level 7 is enabled on Model 15
- Processor check if interrupt level 7 is not enabled on Model 15
- Processor check on Model 12

DA = 1100 and M = 0 specifies drive 1.
DA = 1100 and M = 1 specifies drive 2.
DA = 1101 and M = 0 specifies drive 3; invalid if drive 3 is not installed.
DA = 1101 and M = 1 specifies drive 4; invalid if drive 4 is not installed.

Hex 31, 71, or B1 specifies a load I/O operation. The first hex character in the op code specifies the type of operand addressing for the instruction.

Operation

The processing unit loads the 2 bytes of data contained in the operand into the register specified by the N-code. The operand is addressed by its low-order (higher numbered) storage position.

Program Notes

- An LIO with an N-code of 100 or 110 issued to a busy attachment causes the program to loop on the LIO until the attachment is no longer busy. *Exception:* If the system is equipped with a dual program feature and it is enabled, the processing unit activates the inactive program level.
- LIO does not set any disk status conditions.
- LIO is executed if the addressed drive is executing a seek or recalibrate operation and a read, write, or scan *was not* accepted or provisionally accepted.
- An LIO with an N-code of 100 or 110 is always executed unless the no-op bit is on.

3340/3344 TEST I/O AND BRANCH (TIO)

Op Code (hex)	Q-Byte (binary)	Operand Address	
Byte 1	Byte 2	Byte 3	Byte 4
C1	110x x xxx	Operand 1 address	
D1	110x x xxx	Op 1 disp from XR1	
E1	110x x xxx	Op 1 disp from XR2	

DA M N

N-Code	Condition Tested
000	Not ready/unit check
001	Seek busy
010	Attachment busy
011	Scan hit
100	*Model 12:* Op-end indicator on *Model 15:* Interrupt pending

Any N-code not shown is invalid and causes:
- Program check if interrupt level 7 is enabled on Model 15
- Processor check if interrupt level 7 is not enabled on Model 15
- Processor check on Model 12

DA = 1100 and M = 0 specifies drive 1.
DA = 1100 and M = 1 specifies drive 2.
DA = 1101 and M = 0 specifies drive 3; invalid if drive 3 is not installed.
DA = 1101 and M = 1 specifies drive 4; invalid if drive 4 is not installed.

C1, D1, or E1 specifies a test I/O and branch operation. The first hex character in the op code specifies the type of operand addressing for the instruction.

Operation

The processing unit tests the drive specified by the DA- and M-codes for the condition specified by the N-code. If the condition exists, the program branches to the operand. If the condition does not exist, the program advances to the next sequential instruction.

Resulting Condition Register Setting

This instruction does not affect the condition register.

IAR and ARR Contents After Instruction Execution (Model 15)

If the branch occurred, the IAR contains the branch-to address (from the operand address of the instruction) and the ARR contains the address of the next sequential instruction.

If the branch did not occur, the IAR contains the address of the next sequential instruction and the ARR contains the branch-to address from the operand address of the instruction.

The information stored in the ARR remains there until the next decimal, insert-and-test-characters, branch, or test-I/O instruction is executed.

Program Notes

- Unit check indicates that the addressed disk drive has either a disk drive check status or a common check status outstanding. A common check relates to those sections of the attachment that are shared by all the drives. The usual checks are:

  Command reject
  Invalid track format
  Intervention required
  Track condition check
  Equipment check
  Data check
  No record found
  Write inhibited
  Data overrun
  Command overrun
  Environmental data present
  End of cylinder
  Seek check

  A seek check for the drive not addressed is not indicated. The drive that has the check condition can be determined from the attachment sense bytes.

- Seek busy indicates that the addressed disk drive is performing a seek or recalibrate operation.

- Attachment busy indicates that either the addressed disk drive or attachment:
  a. Is executing a read, write, or scan instruction
  b. Is in the starting phase of the seek operation that requires additional processing unit cycle steal requests
  c. Has provisionally accepted a read, write, or scan instruction for subsequent execution, or
  d. Is currently involved in an IMPL operation.

- Scan hit indicates that the last previous scan operation found the condition specified by the scan command. Scan hit is an indication that is common to all drives; that is, a scan hit on any drive is always indicated to the program, no matter which drive was addressed in the TIO instruction. For example, if a scan command to drive 1 resulted in a scan hit, and drive 2 is addressed by a TIO instruction that specifies testing for a scan hit, a branch occurs.

3340/3344 ADVANCE PROGRAM LEVEL (APL)

Op Code (hex)	Q-Byte (binary)	R-Byte (binary)
Byte 1	Byte 2	Byte 3
F1	110x x xxx	0000 0000

DA M N — R-byte is not used in an APL instruction

N-Code	Condition Tested
000	Not ready/unit check
001	Seek busy
010	Attachment busy
011	Scan hit
100	*Model 12:* Op-end indicator on *Model 15:* Interrupt pending

Any N-code not shown is invalid and causes:
- Program check if interrupt level 7 is enabled on Model 15
- Processor check if interrupt level 7 is not enabled on Model 15
- Processor check on Model 12

DA = 1100 and M = 0 specifies drive 1.
DA = 1100 and M = 1 specifies drive 2.
DA = 1101 and M = 0 specifies drive 3; invalid if drive 3 is not installed.
DA = 1101 and M = 1 specifies drive 4; invalid if drive 4 is not installed.

F1 specifies an APL operation. F as the first hex character in the op code identifies a command-type instruction (that is, an instruction without operand addressing).

Operation

This instruction tests for the conditions specified in the Q-byte.

- Condition present:
  - Systems with Dual Program Feature installed and enabled, activate the inactive program level.
  - Systems without Dual Program Feature installed or with Dual Program Feature installed but not enabled, loop on the advance program level instruction until the condition no longer exists.
- Condition not present: Systems with or without Dual Program Feature access the next sequential instruction in the active program level.

Program Note

For additional information concerning the advance program level instruction, see Chapter 2.

3340/3344 SENSE I/O (SNS)

Op Code (hex)	Q-Byte (binary)	Operand Address	
Byte 1	Byte 2	Byte 3	Byte 4
30	110x x xxx	Operand 1 address	
70	110x x xxx	Op 1 disp from XR1	
B0	110x x xxx	Op 1 disp from XR2	

DA M N

N-Code	Sensed Unit 3340
000	Invalid N-code
001	Invalid N-code
010	Invalid N-code
011	Invalid N-code
100	Disk data address register (DDDR)
101	Status bytes 0 and 1
110	Disk control field address register (DDCR)
111	Diagnostic status bytes

Any N-code not shown is invalid and causes:
Program check if interrupt level 7 is enabled on Model 15
Processor check if interrupt level 7 is not enabled on Model 15
Processor check on Model 12

DA = 1100 and M = 0 specifies drive 1.
DA = 1100 and M = 1 specifies drive 2.
DA = 1101 and M = 0 specifies drive 3; invalid if drive 3 is not installed.
DA = 1101 and M = 1 specifies drive 4; invalid if drive 4 is not installed.

Q-byte bits 0, 1, and 2 = 110 specifies the 3340/3344 attachments as the unit being sensed. Bits 3 and 4 can be any value.

30, 70, or B0 specifies a sense I/O operation. The first hex character in the op code specifies the type of operand addressing for the instruction.

Operation

The attachment transfers 2 bytes of data to the main storage field specified by the operand address. The first byte transferred (the odd-numbered status byte) enters high-numbered storage position in the operand; the other byte enters the low-numbered position of the operand, which is specified by the operand address.

The drive accepts a sense I/O instruction at any time, even through another operation may be in progress when the instruction is issued. See Figure 7-35 for an explanation of the status bits.

Program Notes

- The sense instruction resets the no-op status bit at the end of the sense operation.
- The end-of-cylinder status bit is not valid unless the SNS instruction was issued while the attachment was *not busy* (5445 only).

Byte	Bit	Name	Indicates	Reset By
0	0	Not-ready/unit check, drive 1	*Not-ready* indicates one of the following applies to the indicated drive: (1) powered down, (2) in a disk start transition, or (3) in CE mode. *Unit check* indicates a check occurred while an operation was being performed on the indicated drive, and this condition is not one that is identified by the adapter check status bit (byte 1, bit 7). To recover additional information about the check condition, issue a read diagnostic sense SIO instruction.	Correcting the condition causing the check and resetting the system or issuing another SIO or sense instruction.
0	1	Not-ready/unit check, drive 2		
0	2	Not-ready/unit check, drive 3 (Model 15 only)		
0	3	Not-ready/unit check, drive 4 (Model 15 only)		
0	4	Seek complete on drive 1	A seek or recalibrate operation has been concluded on the indicated drive. If any errors were detected, unit check for the indicated drive will also be posted in this byte. This bit will be on if the adapter no-ops a seek on the indicated drive. *Note:* Seek complete indications are not presented unless interrupts have been enabled.	Issuing a reset seek complete SIO instruction to the indicated drive, or performing a system reset
0	5	Seek complete on drive 2		
0	6	Seek complete on drive 3 (Model 15 only)		
0	7	Seek complete on drive 4 (Model 15 only)		
1	0	CE diagnostic	This bit is reserved for CE diagnostic programming.	CE action
1	1	Scan equal	This bit indicates that a scan equal condition has occurred during execution of a scan instruction. If scan hit is on (returned to a test I/O instruction) scan equal distinguishes between a high and an equal.	Issuing another SIO or sense instruction, or performing a system reset
1	2	Program load selector on removable disk	The program load selector switch is in the DISK1, R1 position. This bit is off for all other program load selector switch positions.	Turning program load selector switch away from DISK 1, R1 position
1	3	Op-end	A read, write, or scan operation has been terminated or an adapter check is pending. If a read, write, or scan instruction results in a no-op condition, this bit will be on. *Note:* Op end is not turned on unless interrupts/op-end indications are enabled. Also, op end is not posted on read extended functional sense or on data module attention control reset instructions.	Issuing a reset op end interrupt SIO or performing a system reset or sense operation
1	4	No-op	An SIO has been accepted by the attachment, but cannot be executed for some reason. This bit is valid for the applicable SIO when op end or seek complete is posted. See *3340/3344 No-Op Conditions* in this section for a list of conditions setting this bit.	An accepted SIO or a system reset
1	5	Data module attention	One of the drives went from not ready to ready, the 3344 R/W or READ switch was moved, or someone pressed the ATTENTION key on one of the drives, causing a recalibrate operation in that drive. The data module attention bit is not active unless interrupts/op-end indications are enabled. *Note:* If any drive has a data module attention outstanding, this bit remains active.	Issuing data module attention control reset SIOs to individual drives until all drives have been reset
1	6	Not used		
1	7	Attachment check (microprocessor halted)	The attachment microprocessor has stopped processing for some reason (possibly an internally detected error) or the microprogram has not been loaded completely.	Loading the microprogram

Figure 7-35. 3340/3344 Disk Drive Status Bytes

IBM 1403 PRINTER

The IBM 1403 Model 5, Model 2, or Model N1 can be attached to the system via an IBM 5421 Printer Control Unit. Each model produces a print line with 132 print positions. The character set can be expanded from 48 characters (basic) to as many as 120 characters by using the universal character set special feature.

Model 2 and Model 5 each require an interchangeable chain cartridge adapter special feature and Model N1 requires an interchangeable train cartridge special feature for installation of the universal character set. Various type fonts, styles, and character arrangements are available.

The printers use a type cartridge with 240 characters. The standard set of graphics, repeated five times on the cartridge, permits the rated throughput of the standard models. Rated throughput, based on a 48-character set with single-line spacing, is:

Model 5	465 lines per minute
Model 2	600 lines per minute
Model N1	1100 lines per minute

Each 1403 has a dual speed carriage, where eight or fewer lines are skipped at a rate of 33 inches per second; larger skips occur at 75 inches per second up to the last eight lines, which are always skipped at 33 inches per second. On System/3, all 1403 document movement is controlled by the stored program.

Polyester film ribbon can be used for optical character recognition and other quality printing applications on the 1403. Model N1 accepts this ribbon without change to the basic machine, but Models 2 and 5 must be equipped with the auxiliary ribbon feeding feature to handle polyester film ribbon.

1403 Not-Ready-to-Ready Interrupt—Model 15 Only

If interrupt level 6 is enabled, the 1403 sends an interrupt request to the system whenever the 1403 goes from a not-ready state to a ready state.

IBM 5203 PRINTER

The IBM 5203 Printer provides hard copy output from the system. This unit is also referred to as the line printer. The printer is available in three models:

Model 1 100 lines per minute
Model 2 200 lines per minute
Model 3 300 lines per minute

The standard print lines is 96 print positions wide. Paper movement is controlled by the program. Interchangeability of type font, styles, or character arrangement is available on all models. All models come equipped with one interchangeable character set cartridge.

A variety of features are available to provide:

- 120 print positions
- 132 print positions
- Dual feed carriage
- Universal character set
- Additional character set cartridges

The printer uses a type cartridge with 240 characters on the cartridge. The standard set of 48 characters, repeated five times on the cartridge, permits the rated throughput of 100, 200 or 300 lines per minute. The character set can be expanded from 48 to as many as 120 characters by using the universal character set special feature. However, when this feature is used, throughput will decrease depending on the text being printed.

5203 Operational Limitation on Model 10

Because of its data transfer rate requirements, the 5203 is subject to data overrun when its operations are overlapped with other devices in certain system configurations. This condition is not detected by the 5203 and may result in loss of data. Refer to *Channel Limitations on Model 10 Configurations* in Chapter 1, for allowable overlapped device configuration that will not cause overrun in the system.

Print Considerations for the Dual-Feed Carriage

When dual-feed carriage is installed, carriage instructions are referenced to the left and right carriages. When the dual-feed carriage feature is not installed, only the left carriage commands are effective.

When dual-feed carriage is used, a minimum of 17 positions are lost between the last character on the left form and the first character on the right form (assuming carrier strips are used).

For best print quality in dual-feed-carriage systems, the forms thickness should be the same in both carriages.

LINE PRINTER OPERATIONS

Initialization

Before any print operation can be performed successfully, the system must be initialized as follows:

Print Image and Line Printer Image Address Register (LPIAR)

The line printers use interchangeable character sets. The correct character train or chain must be installed in the printer, and its character set image (the sequence of print characters as they appear on the train or chain) must be stored in an I/O area of main storage. This area is called the line printer image area.

The print image must be stored and its address must be loaded after every power down condition and every time a different character set is used for a new application. The programmer selects the line printer image area, stores chain or train image in this area, and loads the address (the left-most byte) of the line printer image area into a local storage register called the line printer image address register.

Print Data Field and Line Printer Data Address Register

Before performing any print operations, the program must load the address of the leftmost byte of the print data area into the line printer data address. This field serves as an output buffer for data to be printed on a single line of the form.

Forms Length and Forms Length Register

Line printers use continuous forms, which vary in length (from top to bottom) from job to job. The program must specify the forms length for each job by loading the forms length register (a local storage register) before the job is run (see *1403/5203 Load I/O [LIO]* in this section).

Printing

Start-I/O instructions control printer operations. Test-I/O and sense instructions test printer status to establish program branch decisions. During processing operations, the program must format one print line at a time in the print data area. Each position on the line to be printed must be stored in the associated print data area before the program issues the print instruction. That is, blanks must be stored in print positions to remain unprinted, and appropriate characters must occupy all other positions of the line printer data area.

Forms Control

The maximum length of a form is 14 inches (112 line spaces at eight lines per inch or 84 line spaces at six lines per inch spacing).

Forms can be moved at either six lines per inch or eight lines per inch. Spacing can be performed in increments of 0, 1, 2, or 3 line spaces. Skips can be any length up to the value established in the forms length register by the load I/O instruction.

Forms movement is entirely controlled by start I/O instructions. Instructing a line printer to skip to a line that exceeds the value in the forms length register has the following results:

- *1403:* The printer skips to a line on the next form that is equal to the difference between the forms length value and the line number to which the 1403 is programmed to skip.
- *5203:* The attachment posts a check condition.

Detection of Printing Line Location

It is often necessary to determine the current print line location to perform forms overflow control, heading control, and other controlled forms movement. The program can issue a sense I/O instruction to determine the current line location.

When the last line of the form is in the print position and a 1403 load I/O instruction is issued to change the forms length, the line counter will indicate an incorrect line count.

Print Area Restrictions

The line printer data area and the line printer image area in storage must occupy certain regions within 256-byte boundaries. That is, the high-order byte of the address can contain any value within the range of addresses of the particular system, but the low-order byte must contain particular addresses. The particular addresses required are arranged such that the line printer data area and the line printer image area can (but are not required to) occupy regions within the same 256-byte area of storage. The following requirements must be met:

1. The 48-character set image must be in the 48 bytes having low-order address bytes of hex 00 through 2F.
2. The 120-character set image must be in the 120 bytes with low-order address bytes of hex 00 through 77.
3. The line printer data for 96 print positions (5203 only) must occupy the 96 bytes with low-order address bytes of hex 7C through DB.
4. The line printer data for 120 print positions (5203 only) must occupy the 120 bytes with low-order address bytes of hex 7C through F3.
5. The line printer data for 132 print positions must occupy the 132 bytes with low-order address bytes of hex 7C through FF.

The line printer data area in storage beginning at location xx7C corresponds character for character to the print line beginning at print position 1.

1403/5203 START I/O (SIO)

Op Code (hex) Byte 1	Q-Byte (binary) Byte 2	R-Byte (binary) Byte 3
F3	1110 x xxx	xxxx xxxx

DA M N Control Code

N-Code	Operation	Bits 0123 4567	Function
000	Line space only	0000 0000	Line-space 0 lines
		0000 0001	Line-space 1 line
010	Print and linespace	0000 0010	Line-space 2 lines
		0000 0011	Line-space 3 lines
		All other control codes inhibit line-spacing when N = 000 or 010.	
011[2]	Interrupt control for 1403 on Model 15	1xx0 0000[1]	Enable op end interrupt
		0xx0 0000[1]	Disable op end interrupt
		x1x0 0000[1]	Reset interrupt caused by no-op condition or by printer buffer going not-busy.
		xx10 0000[1]	Reset interrupt caused by no-op condition or by carriage going not-busy.
100	Skip only	0000 0000 through 0111 0000	Skip the decimal line number specified by the binary control code. This line number can be anything from 0 through 112. (See *Program Notes.*)
110	Print and skip		

Any N-code not shown is invalid and causes:
- Program check if interrupt level 7 is enabled, on Model 15
- Processor check if interrupt level 7 is not enabled, on Model 15
- Processor check on Models 8, 10, and 12

1403 M-bit: 0 = normal print operations
1 and R-byte of hex 80 = CE diagnostic mode
1 and R-byte not hex 80 = invalid M-bit, causing processor check

5203 M-bit: 0 = left carriage control in a dual-feed carriage system
1 = right carriage control in a dual feed carriage system
1 = invalid M-bit for systems without dual-feed carriage, causing processor check

Hex E specifies line printer as the device being controlled.

F3 specifies a start I/O operation. F, as the first hex character in the op code, specifies a command-type instruction (that is, an instruction with no operand addressing).

---

[1]x = can be 1 for multiple-function control.
[2]Invalid N-code for Models 8, 10, and 12.

Operation

This instruction can initiate forms movement and/or printing or can initiate interrupt control functions. If printing is specified, the data contained in the printer data area of storage is printed as a single line, beginning at the address specified in the line printer data address register. Unprintable characters and coded blanks (hex 40) print as blanks. Unprintable characters set a testable indicator and remain in the data area. All positions in which characters are printed are set to hex 40. If forms movement is specified, the printer spaces or skips as specified by the R-byte.

Program Notes

- If the skip-to number exceeds the number of the last line on a form, (1) the 1403 skips to a line equal to the specified destination less the forms length on the next form or (2), a check condition occurs on the 5203. A skip to a line less than that at which the carriage is located results in a skip to the specified line on the following page. A skip to the line at which the carriage is located results in no carriage motion.

- A parity error detected by the attachment results in a processor check stop and lights the DBO parity check light. The attachment sets the no-op status bit if a device error exists when start I/O is executed.

- If the printer is busy or intervention is required when the start I/O instruction is executed, the program loops on the start I/O instruction if the Dual Program Feature is not installed, or automatically advances the program level if the Dual Program Feature is installed.

- In a system using a 5203 with a dual-feed carriage, a control instruction for a specific carriage will be accepted if that carriage is not busy, but execution is delayed until any printing from that or a previous instruction is completed. Forms motion of both carriages can be accomplished by giving a print and forms motion instruction to one carriage followed by a forms motion instruction to the other carriage.

- The no-op indicator indicates that the last SIO instruction issued was accepted but was not executed because of a printer check condition. The no-op indicator is reset by a system reset, a system check reset, or an SNS instruction.

- The first TIO for ready instruction issued after the no-op bit is set causes the program to branch. If the no-op bit is on, the program should issue the last SIO instruction used, because no data has been lost.

- If a printer buffer busy turns off or a carriage busy turns off during an interval that interrupts are not enabled, interrupt requests will occur when interrupts are enabled. If this is not desired, issue an SIO specifying reset/enable (control code = hex E0).

Op-End Interrupt Request (Model 15)

The 1403 attachment presents an op-end interrupt request to the CPU at the end of the CPU instruction during which one of the following conditions has occurred:

- Printer buffer went from busy to not busy[1]
- Carriage went from busy to not busy[1]
- SIO instruction was not executed, but no-op bit was set because of an equipment check.

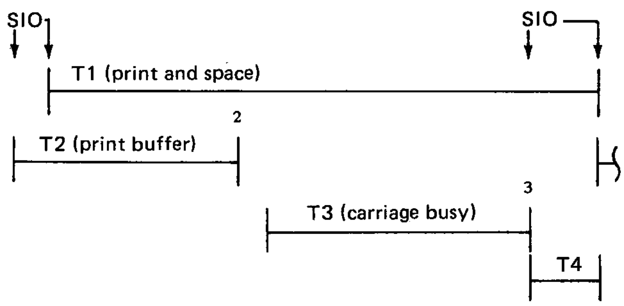
---

[1]The 1403 attachment presents an immediate op-end interrupt request to the CPU whenever interrupt is enabled (after having been disabled) while the printer buffer is not busy and the carriage is not busy. To avoid this interrupt request, use a reset/enable command (control code = hex E0). Test for interrupt pending conditions by issuing a TIO instruction with a Q-byte of hex E3. To determine the condition causing the interrupt request, test as follows:

Instruction	Q-Byte	Tested Condition
TIO	X'E2'	Print buffer busy
TIO	X'E4'	Carriage busy
SNS	X'E3'	No-op instruction

[2]Print buffer not busy op-end interrupt
[3]Carriage not busy op-end interrupt

1403 Op-End Interrupt Timings (Nominal Times)



1403 \ Time (ms)	T1	T2	T3	T4
N1	54	36.9	20.4	0.5
2	100	83.8	20.4	1.0
5	129	111.7	21.4	1.3

1403/5203 TEST I/O AND BRANCH (TIO)

Op Code (hex)	Q-Byte (binary)	Operand Address	
Byte 1	**Byte 2**	**Byte 3**	**Byte 4**
C1	1110 x xxx	Operand 1 address	
D1	1110 x xxx	Op 1 disp from XR1	
E1	1110 x xxx	Op 1 disp from XR2	

DA M N

N-Code	Condition Tested
000	Not ready/check
001	*1403:* With M-bit = 1 specifies test for diagnostic mode off. With M-bit = 0 — Invalid N-code. *5203:* Invalid N-code.
010	Print buffer busy
011[1]	Interrupt pending
100	Carriage busy
110	Printer busy

Any N-code not shown is invalid and causes:
- Program check if interrupt level 7 is enabled on Model 15
- Processor check if interrupt level 7 is not enabled on Model 15
- Processor check on Models 8, 10, and 12

5203 M-bit: Used for dual carriage testing.
- 0 = test of left carriage on printer equipped with dual-feed carriage; must be used on 5203 without dual-feed carriage.
- 1 = test of right carriage on printer equipped with dual feed carriage. Invalid M-bit on 5203 without dual feed carriage, causing processor check.

1403 M-bit: Used for printer testing.
- 0 = tests other than CE diagnostic tests
- 1 with N-code of 001 = test for diagnostic mode off
- 1 with N-code not 001 is invalid

Hex E specifies line printer as the tested device.

E1, D1 or E1 specifies a test I/O and branch operation. The first hex character in the op code specifies the type of operand addressing for the instruction.

[1] Invalid N-code for Models 8, 10, and 12.

Operation

The processing unit tests the 1403 or 5203 for any condition specified by the N-code. If one of the tested conditions exists, the program branches to the operand address. If no tested condition exists, the program proceeds with the next sequential instruction.

Resulting Condition Register Setting

This instruction does not affect the condition register.

Program Notes

- Not-ready/check condition becomes active any time the print becomes not ready for any reason. It becomes inactive when the reason for the not ready condition is removed.
- Print buffer busy condition becomes active when the printer accepts a start I/O instruction that specifies printing. It becomes inactive when the line has been printed but before carriage motion stops.
- Carriage busy condition becomes active when the printer accepts a start I/O instruction that specifies carriage motion. It becomes inactive when carriage motion stops.
- Printer busy condition becomes active as soon as the printer accepts any start I/O instruction and becomes inactive when the instruction has been completely executed.
- A parity error detected by the attachment results in a program check or a processor check with the DBO parity check light on.

IAR and ARR Contents After Instruction Execution—Model 15

- If the branch occurred, the instruction address register contains the branch-to address and the address recall register contains the address of the next sequential instruction.
- If the branch did not occur, the instruction address register contains the address of the next sequential instruction and the address recall register contains the branch-to address.

1403/5203 ADVANCE PROGRAM LEVEL (APL)

Op Code (hex)	Q-Byte (binary)	R-Byte (binary)
Byte 1	Byte 2	Byte 3
F1	1110 x xxx	0000 0000

DA M N — R-byte is not used in an APL instruction.

N-Code	Condition Tested
000	Not-ready/check
010	Print buffer busy
011	Interrupt pending
100	Carriage busy
110	Printer busy

Any N-code not shown is invalid and causes:
- Program check if interrupt level 7 is enabled on Model 15
- Processor check if interrupt level 7 is not enabled on Model 15
- Processor check on Models 8, 10, and 12

5203 M-bit refers to the dual-feed carriage feature. When the M bit is 0, the left carriage can be tested; when the M bit is 1, the right carriage can be tested. If an M bit of 1 is used when the dual feed carriage is not installed, a processor-check stop results with an invalid device address indication.

1403 M-bit of 0 specifies a printer condition to be tested. An M bit of 1 with an N-code of 001 specifies a test for diagnostic mode off; an M bit of 1 with any other N-code is invalid.

Hex E specifies line printer as the tested device.

F1 specifies an APL operation. F as the first hex character in the op code identifies a command-type instruction (that is, an instruction without operand addressing).

Operation

This instruction tests for the conditions specified in the Q-byte.

- Condition present:
  - Systems with Dual Program Feature installed and enabled, activate the inactive program level.
  - Systems without Dual Program Feature installed or with Dual Program Feature installed but not enabled, loop on the advance program level instruction until the condition no longer exists.
- Condition not present: Systems with or without Dual Program Feature access the next sequential instruction in the active program level.

Program Notes

- Not-ready/check condition becomes active any time the printer becomes not ready for any reason. It becomes inactive when the reason for the not ready condition is removed.
- Print buffer busy becomes active when the printer accepts a start I/O instruction that specifies printing. It becomes inactive when the line is printed but before carriage motion stops.
- Carriage busy becomes active when the printer accepts a start I/O instruction that specifies a carriage operation. It becomes inactive when carriage motion stops.
- Printer busy becomes active as soon as the printer accepts any start I/O instruction and becomes inactive when the instruction has been completely executed.
- Byte 3 of this instruction is not used. Care should be exercised in punching program cards to ensure that the op code byte for the following instruction is not inadvertently punched in the column that should be occupied by byte 3 of this instruction.
- For additional information concerning the advance program level instruction, see Chapter 2.

1403/5203 LOAD I/O (LIO)

Op Code (hex)	Q-Byte (binary)	Operand Address	
Byte 1	Byte 2	Byte 3	Byte 4
31	1110 x xxx	Operand 1 address	
71	1110 x xxx	Op 1 disp from XR1	
B1	1110 x xxx	Op 1 disp from XR2	

DA M N

N-Code	To Be Loaded
000	Forms length register
100	Line printer image address register
110	Line printer data address register

Any N-code not shown is invalid and causes:
- Program check if interrupt level 7 is enabled on Model 15
- Processor check if interrupt level 7 is not enabled on Model 15
- Processor check on Models 8, 10, and 12

1403 M-bit: 0 specifies a normal processing mode function (should be used).
1 specifies a CE diagnostic mode function.
5203 M-bit: Has no significance; should be 0.

Hex-E specifies line printer as the device whose registers are to be loaded.

31, 71, or B1 specifies a load I/O operation. The first hex character in the op code specifies the type of operand addressing to be used for the instruction.

Operation

The processing unit loads the 2 bytes of data contained in the operand into the register specified by the N-code. The operand is addressed by its low-order (higher numbered) storage position. If the printer no-op bit is on, the processing unit bypasses this instruction and immediately accesses the next sequential instruction. If the addressed register is busy, the program loops on the LIO instruction until the register becomes not busy.

For a load forms length register operation, the effective address byte (low-order byte of the addressed field) contains the forms length for the right carriage; the effective address minus 1 byte contains the forms length for the left carriage. (Forms length is determined by measuring the form from top to bottom, and multiplying the length in inches by the number of lines to be printed per inch. For example, an 11-inch form to be printed at six lines per inch spacing has a forms length of 66. The same form printed at eight lines per inch spacing has a forms length of 88.) *If the printer is not equipped with the dual carriage,* the effective address byte can contain any data you wish to store, for this byte is not used for the instruction.

Program Notes

- Determining end of page is a programming function.
- Issuing a load I/O instruction to change the forms length when positioned on the last line of the form will cause the line counter to be incorrect.

1403/5203 SENSE I/O (SNS)

Op Code (hex)	Q-Byte (binary)	Operand Address	
Byte 1	Byte 2	Byte 3	Byte 4
30	1110 0 xxx	Operand 1 address	
70	1110 0 xxx	Op 1 disp from XR1	
B0	1110 0 xxx	Op 1 disp from XR2	

DA M N

Information Moved

N-Code	Byte 1	Byte 2
000	5203 right carriage line location	5203 left carriage line location
	1403 character count	1403 carriage line location
001	5203 chain character counter	5203 incrementing factor of LPDAR
	1403 invalid	1403 invalid
010	Printer timing—byte 1	Printer timing—byte 2
011	Printer check status—byte 1	Printer check status—byte 2
100	LPIAR—low byte	LPIAR—high byte
110	LPDAR—low-order byte	LPDAR—high-order byte

Any N-code not shown is invalid and causes:
- Program check if interrupt level 7 is enabled on Model 15
- Processor check if interrupt level 7 is not enabled on Model 15
- Processor check on Models 8, 10, and 12

M-code is not used; should be 0.

Hex E specifies line printer as the addressed device.

30. 70, or B0 specifies a start I/O operation. The first hex character in the op code specifies the type of operand addressing for the instruction.

Operation

The CPU transfers 2 bytes of data specified by the N-code to the main storage field specified by the operand address. The first byte transferred enters the effective address (the operand address), the second byte enters the effective address minus 1. The sense I/O instruction is executed even if the printer is busy. Status bits are described in Figure 8-1.

Byte	Bit	Name	Indicates	Reset By
1	0	Chain sync check	Incorrect characters were printed and correct characters from printer data area of storage were replaced with blanks.	5203 printer start key, 1403 or processing unit check reset key
1	1	5203 incrementer sync check 1403 not used	Incrementing hammer unit is out of sync with the printer attachment or a roller clutch failed in the incrementer cam.	
1	2	5203 thermal check 1403 not used	Something overheated in the hammer unit. (Call the CE if successive thermal checks occur.)	Processing can continue as soon as the hammer unit cools
1	3	5203 not used 1403 echo check of set address	Last line of printing may be incorrect. (Reexecuting the last SIO reprints the last line without loss of any data.)	1403 or processing unit check reset key
1	4	5203 not used (always on) 1403 interlock check	A cover or printer interlock is open.	Closing open interlock and pressing 1403 or processing unit check reset key
1	5	48-character set	48-character set is installed in 1403.	Removing character set
1	6	Unprintable character	Program sent character to printer that printer is not capable of printing with character set installed.	Next SIO issued or next system reset
1	7	CE sense bit (diagnostic)		
2	0	Carriage sync check	The carriage has spaced or skipped more than programmed due to the loss of synchronism between the carriage and its attachment.	5203 printer start key, 1403 or processing unit check reset key
2	1	5203 carriage space check 1403 not used	Same as carriage sync check.	
2	2	Forms jam check	Forms crumpling or tearing in forms tractor area. (The remainder of the last destroyed form will print on the new form.)	5203 printer start key, 1403 or processing unit check reset key
2	3	5203 incrementer failure check 1403 print data check	Incrementer hammer unit failed to move. Reexecuting the last SIO prints the rest of the line without any loss of data (5203); a parity error during data access from the print buffer during a print operation (1403).	5203 printer start key, 1403 or processing unit check reset key
2	4	CE sense bit latched (diagnostic)		
2	5	Print check (hammer echo check)	Print hammer did not respond properly to a print signal so data was not printed dependably. Reexecuting the last SIO reprints the last line without any loss of data.	5203 printer start key, 1403 or processing unit check reset key
2	6	Print check (any hammer on check)	Hammer did not return to its proper position after striking the character slug.	5203 printer start key, 1403 or processing unit check reset key
2	7	No-op	The last SIO issued was not performed because the SIO specified a function; the printer could not perform.	5203 printer start key, 1403 or processing unit check reset key or next SIO issued.

Note: Byte 1 is stored at operand 1 address; byte 2 is stored at operand 1 address minus 1.

End of forms. This check does not have a status bit. It is indicated by the I/O attention and forms lights.

Interlock conditions. These conditions do not have status bits. They are indicated by the I/O ATTENTION and INTERLOCK lights. On the 1403, interlock conditions are indicated by the I/O ATTENTION and the PRINT CHECK light or the FORMS CHECK light. An internal indicator panel shows the appropriate interlock condition.

Figure 8-1. Line Printer Status Bytes

Dual Program Feature (Models 8, 10, and 12)

Although the dual program feature, the interval timer, and the not-ready-to-ready restart logic reside in the processing unit, each is programmed with input/output instructions as though it were an I/O unit.

The Dual Program Feature lets the system execute two independent programs on a time-sharing basis. That is, it allows the processing unit to transfer to a different program when the current program must wait for completion of an I/O operation.

Two independent object programs can reside in storage simultaneously. This uses the high performance capabilities of the processing unit rather than forcing it to wait for completion of the execution by active I/O devices.

The transfer from one program level to the other is called program level advance. Program level advance can be either automatic or program-controlled. Unlike interrupt, program level advance does not require that index registers 1 and 2 and the program status register be stored, because separate index registers, instruction registers, address recall registers, and program status registers are provided for each program level.

An automatic program level advance occurs when:

1. Operation on one program level is instructed to halt.
2. An I/O device is instructed to operate when the device requires operator attention.
3. An I/O device is instructed to operate when the device is already performing an operation.

A program-controlled program level advance is accomplished by issuing an APL instruction.

Program Note: After a system reset, a program level advance from program level 1 to program level 2 will initialize the condition register to the high condition.

Because one program can finish operating before the other, and thus require a new program to be entered while one of the old programs is running, it is the responsibility of the supervising program to ensure that the two programs do not use the same I/O devices or overlapping storage areas.

The following instructions must be incorporated in the loader/supervisor program for dual program control:

DUAL PROGRAM START I/O (SIO)

Op Code (hex)	Q-Byte (binary)	R-Byte (binary)
Byte 1	Byte 2	Byte 3
F3	0000 0000	0000 0xxx

Control Code

Bit	Operation
0	Reserved
1	Reserved
2	Reserved
3	Reserved
4	Reserved
5	Enable dual program mode when bit is 1; disable dual program mode when bit is 0
6	Enable interrupt level 0 (system control panel interrupt key) when bit is 1; disable interrupt level 0 when bit is 0
7	Reset interrupt request 0

Hex F3 specifies a start I/O operation. Hex F as the first digit in the op code signifies that the instruction is a command-type instruction (that is, no operand addressing is used).

Operation

This instruction controls the dual program mode of operation and the dual program interrupt level. The control code specifies the operation to be performed.

The start I/O instruction to enable or disable dual program mode provides programmed control over the system's ability to execute a program level advance. Enabling the dual program mode allows both the automatic and the programmed advance of the program levels to occur. Disabling dual program mode inhibits all program level advances and transforms them into wait operations. This instruction can be issued in either program level or in any interrupt level and will enable or disable all program level advances until another enable or disable instruction is given.

Program Notes

- Program level advances are not executable in an interrupt level. Unconditional program level advances result in no-op operations, and conditional program level advances result in wait operations.

- To enable interrupt level 0, bits 5 and 6 of the control code must both be present. Interrupt level 0 cannot be enabled unless dual program mode is enabled.

DUAL PROGRAM TEST I/O AND BRANCH (TIO)

Op Code (hex)	Q-Byte[1] (binary)	Operand Address	
Byte 1	Byte 2	Byte 3	Byte 4
C1	0000 0 xxx	Operand 1 address	
D1	0000 0 xxx	Op 1 disp from XR1	
E1	0000 0 xxx	Op 1 disp from XR2	

DA M N

Bit 5 defines the program level to be operated on:
- 0: Program level 1
- 1: Program level 2

Bits 6	7	Condition
0	0	Cancel program level
0	1	Load program level from MFCU
1	1	Load program level from alternate device
1	0	Load program level from printer-keyboard

C1, D1, or E1 specifies a test I/O and branch operation. The first hex character in the op code specifies the type of operand addressing for the instruction.

Operation

This instruction tests the setting of the dual program control switch on the system control panel. The N-code specifies the condition to be tested.

Interval Timer—Model 15 Only

This feature is a 3-byte binary counter that is loaded by a load I/O instruction, stored by a sense instruction, and controlled by start I/O instructions. The counter can store a maximum binary value equal to decimal 16,777,215. During timer operation, this value decreases by 1 each 3.3 ms (timer cycle rate is 300 Hz) until the value has reached negative 0; then the value increases by 1 each 3.3 ms. The total cycle time for the timer is about 15.5 hours.

The timer generates an interrupt on level 6 whenever the value in the timer changes from positive (including 0 as a positive number) to negative. The timer revolution is 3.33 ms. That is, if the timer is set to hex 000001, an interrupt occurs 1.67 to 5.00 ms after the timer is started. The accuracy of the timer is 0.075% of the interval timed or 3.33 ms, whichever is greater.

The timer does not take cycle steals and in no way affects system burden. The value in the timer is always current, as the timer is not dependent on the system but has its own clock. (The timer is not stopped when the system is at a halt or is stopped, provided the system clock is running. Therefore, by using a program routine, the interval timer can be used as a time-of-day clock.)

Figure 9-1 is a schematic of the timer counter bit and byte identification.

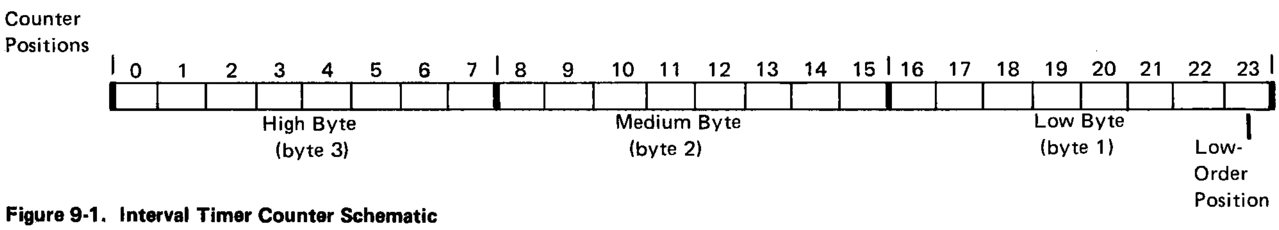


Figure 9-1. Interval Timer Counter Schematic

Not-Ready-To-Ready Interrupts—Model 15 Only

The not-ready-to-ready interrupt logic resides in the processing unit, but is enabled, disabled, and reset using a start I/O instruction that has the same device address and interrupt level (level 6) as the interval timer; it is tested by a test I/O instruction that has the same device address as the interval timer. The not-ready-to-ready interrupt request indicates that one of the following units has gone from a not-ready state to a ready state:

1403
1442
2501
2560
5424

When a level 6 interrupt is indicated, the program can test to determine whether the interrupt was an op end interrupt for the interval timer or a not-ready-to-ready interrupt request. If the result shows a not-ready-to-ready interrupt request, individual units can be tested for a ready condition.

NOT-READY-TO-READY AND INTERVAL TIMER START I/O (SIO)

Op Code (hex)	Q-Byte (binary)	R-Byte (binary)
Byte 1	Byte 2	Byte 3
F3	0000 1 00x	0000 0xxx

DA M N — Control Code

Bit	Meaning if N=000	Meaning if N=001
0-4	Not used; should be 0	Not used; should be 0
5	0 = Stop timer 1 = Start timer	Not used; should be 0
6	0 = Disable interrupt 1 = Enable interrupt	1 = Reset interrupt request
7	1 = Reset interrupt request	0 = Disable interrupt 1 = Enable interrupt

N=000 specifies interval timer control
N=001 specifies not ready to ready interrupt control
Any other N code is invalid and causes:
Program check if interrupt 7 is enabled
Processor check if interrupt 7 is disabled

00001 specifies the interval timer or not-ready-to-ready interrupt logic as the addressed unit.

Hex F3 specifies a start I/O operation. Hex F as the first digit in the op code signifies that the instruction is a command-type instruction (that is, no operand addressing is used).

Operation

The attachment performs the functions specified by the control code.

Program Notes

- After being started by a start I/O instruction, the timer is decreased by one every 3.3 ms until stopped by a start I/O stop timer command, an LIO instruction, a system reset, or by stopping the system clock. Note that the timer does not stop counting when changing from positive to negative value.
- The interval timer runs while the system is at a halt or is stopped with the system clock running. If the system clock stops, the timer stops.
- Once the timer stops for any reason, it does not resume operation until an SIO start timer command is issued.
- The timer always accepts an SIO instruction.
- If the timer is not active, an SIO start instruction may decrease the counter by one immediately. Therefore, the time at which an interrupt occurs may be affected significantly if the timer is started and stopped a number of times within a certain interval.

Interrupts

The timer operates on interrupt level 6. It presents an interrupt request to the processing unit whenever interrupt is enabled and the timer value changes from positive (including zero) to negative.

If interrupt is not enabled (disabled) when the timer changes from positive to negative, the interrupt is lost.

An SIO to disable interrupt level 6 does not reset an interrupt in process, but does prevent any subsequent interrupt from occurring.

The timer does not stop counting when an interrupt occurs.

The not-ready-to-ready interrupt logic also operates on interrupt level 6. Whenever not-ready-to-ready interrupts are enabled, any 1403, 1442, 2501, 2560, or 5424 unit going from not ready to ready initiates a CPU program interrupt. The program must test to determine whether the timer or the not-ready-to-ready interrupt logic interrupted the program; then, if the latter, test to determine which I/O unit caused the interrupt.

NOT-READY-TO-READY INTERRUPT PENDING TEST I/O AND BRANCH (TIO)

Op Code (hex)	Q-Byte[1] (binary)	Operand Address	
Byte 1	Byte 2	Byte 3	Byte 4
C1	0000 1 001	Operand 1 address	
D1	0000 1 001	Op 1 disp from XR1	
E1	0000 1 001	Op 1 disp from XR2	

DA M N

N-Code	Condition Tested
001	Not-ready-to-ready interrupt pending

DA M: Specifies the not-ready-to-ready interrupt logic as the unit being tested.

C1, D1, or E1 specifies a test I/O and branch operation. The first hex character in the op code specifies the type of operand addressing for the instruction.

[1]Any Q-byte not shown causes:
Program check if interrupt level 7 is enabled
Processor check if interrupt level 7 is disabled

Operation

The processing unit tests the not-ready-to-ready interrupt logic for an interrupt pending condition. If an interrupt is pending, the program branches to the address specified by the operand 1 address in the instruction. If no interrupt is pending, the program proceeds with the next sequential instruction.

Resulting Condition Register Setting

This instruction does not affect the condition register.

INTERVAL TIMER LOAD I/O (LIO)

Op Code (hex)	Q-Byte (binary)	Operand Address	
Byte 1	Byte 2	Byte 3	Byte 4
31	0000 1 00x	Operand 1 address	
71	0000 1 00x	Op 1 disp from XR1	
B1	0000 1 00x	Op 1 disp from XR2	

DA M N

N-Code **Unit Loaded**

000 Timer low byte. (Although 2 bytes of data are taken from storage, the addressed byte is not used. The byte stored at the operand address minus 1 is loaded into counter positions 16 through 23.)

001 Timer high and medium bytes. (The byte stored at the operand address is loaded into counter positions 8 through 15; this byte is called the timer medium byte. The byte stored at the operand address minus 1 is loaded into counter positions 0 through 7; this byte is called the timer high byte.)

Any other N-code is invalid and causes:
- Program check if interrupt level 7 is enabled
- Processor check if interrupt level 7 is not enabled

00001 specifies the interval timer as the addressed device.

Hex 31, 71, or B1 specifies a load I/O operation. The first hex character in the op code specifies the type of operand addressing for the instruction.

Operation

The processing unit loads the data stored in the field specified by the operand address into the interval counter positions specified by the N-code.

Program Note

The program must issue two LIO instructions, each with a unique N-code, to load the interval timer counter. The first of these instructions stops the timer. After the timer is completely loaded (both LIOs have been issued) the timer can be started with an SIO start timer command.

INTERVAL TIMER SENSE I/O (SNS)

Op Code (hex)	Q-Byte (binary)	Operand Address	
Byte 1	Byte 2	Byte 3	Byte 4
30	0000 1 00x	Operand 1 address	
70	0000 1 00x	Op 1 disp from XR1	
B0	0000 1 00x	Op 2 disp from XR2	

DA M N

N-Code	Timer Data Being Sensed
000	Timer low byte. (The timer attachment provides the processing unit with binary 0 byte for the data to be stored in the addressed storage location. The timer low byte enters the operand address minus 1.)
001	Timer medium byte and timer high byte. (Timer medium byte enters the position specified by the operand address. Timer high byte enters the operand address minus 1 position.)

Any other N-code is invalid and causes:
- Program check if interrupt level 7 is enabled
- Processor check if interrupt level 7 is not enabled

00001 specifies the interval timer as the addressed device.

Hex 30, 70, or B0 specifies a sense I/O operation. The first hexadecimal digit in the op code signifies the type of operand addressing being used.

Operation

The processing unit stores the data specified by the N-code into the storage data field specified by the operand address. The operand is addressed by its low-order (rightmost or higher-numbered) position; it is a 2-byte field.

Program Notes

- The program must issue two sense instructions to store the contents of the interval timer. The first sense instruction issued must have an N-code of 000, the second must have an N-code of 001 to store the data. This sequence allows the attachment to store the entire contents of the timer counter into a 24-position buffer upon recognition of the N-code of 000. This prevents the possibility of presenting erroneous timer data if the timer should be decreased between the two SNS instructions.

- A Q-byte of hex 00 causes the data set up in the address/data switches on the system control panel to be stored in the field specified by the operand 1 address of the instruction.

This chapter describes communications adapters (attachments that can be installed on the system to permit I/O communications with devices usually serving as terminals); the chapter also presents the I/O instructions that must be issued to control these devices via their attachment features. Although all of the devices discussed are considered terminals, some are used in a local, rather than remote, environment.

Note: In this chapter, unless otherwise stated, the term adapter refers to all communications features.

BSCA, BSCC, AND ICA

BSCA, BSCC, and ICA are special features that can be attached to various models of System/3 (see *System Configurations By Model* in Chapter 1). The BSCA and ICA features provide the system with the ability to function as a point-to-point or multipoint control terminal. Operation is half-duplex, synchronous, and serially by bit, over either nonswitched or switched voice grade or better two-wire, four-wire, or wide band communication facilities.

The BSCC provides the system with the ability to function as a multipoint control station only. The BSCC also provides one or two BSC lines designed to communicate with remote terminals, workstations, and other systems. These lines operate at line speeds ranging from 600 bits per second to 9,600 bits per second. Both lines can be operated simultaneously as viewed by the system. Operation is half-duplex, serially by bit, and serially by character over nonswitched two-wire or four-wire voice grade facilities. Each line functions independently of the other and each line can be configured differently using the available subfeatures and options.

Operation of the communications adapter is fully controlled by a combination of System/3 stored program instructions and logical responses to line control characters. With a communications adapter installed, the system can both transmit and receive during a single communication, although half-duplex operation prevents simultaneous transmission and reception of data.

Point-to-Point Communications Networks

The BSCA/ICA functions in either a switched or nonswitched point-to-point network. Normally, contention cannot occur because the called station must be made ready to receive before a call can be completed. However, a 2-second timeout can be programmed to resolve any contention situations that may occur.

System/3 can be designated, by programming, as either the primary or secondary station.

Note: BSCC is not supported as a point-to-point station.

Multipoint Communications Networks

IBM supports System/3 both as a tributary station and as a control station on a multipoint network.

Note: BSCC is supported as a multipoint control station only.

Data Rates

The first BSCA can operate at various rates between 600 bits per second (baud) and 50,000 bits per second. The second BSCA and BSCC on a single system operate at a maximum rate of 9,600 bits per second. The ICA can operate at various data rates between 600 bits per second and 8,000 bits per second. The customer selects the data rate to be used, and the ICA is equipped with an appropriate interface as a no-charge selective feature. Interconnected units must operate at the same data rate.

Data Set Interface

The data set interface modifies the BSCA/ICA/BSCC for operation on voice grade communications channels. This interface makes possible data rates between 600 and 9,600 bits per second, provided the appropriate data set is installed. (For information about acceptable data sets or their equivalents consult your IBM sales representative.)

Local Attachment Feature Interface

The BSCA and BSCC can be equipped with an EIA Local Attachment Feature that allows it to communicate with a device like the IBM 3270 Information Display System located in the immediate area (without the use of a data set). With this feature attached, the data rates are as follows:

Feature	Data Rates
BSCA	2,400, 4,800, or 8,000 bps
BSCC	1,200, 2,000, 2,400, 4,800, 7,200, or 9,600 bps
ICA	2,400 or 8,000 bps

Data Station Interface

The data station interface modifies the first BSCA for operation on wide band communications channels at data rates between 19,200 and 50,000 bits per second. (For information about acceptable data sets or their equivalents consult your IBM sales representative.)

Data Sets (Modems)

The data set receives the data serially by bit and serially by character from the communications line during receive operations and presents the bits to the communications adapter. During transmit operations the communications adapter receives characters from storage serially, then makes them available serially by bit, serially by character to the data set. The data set puts each bit on the communications line as soon as it receives the bit from the BSCA, BSCC, or ICA.

At the time he places his order for the adapter, the customer should understand the requirements of the data set being used.

Transmission Rate Control

A timing device called a clock controls the rate at which data is transmitted and received. For the data set interface, clocking is furnished either as a special feature for the adapter or else by the data set, depending on which type of data set is selected. For the data station interface, the data set must furnish the clock. Clocking is furnished as part of the feature when the EIA local attachment feature is installed.

Transmission Codes

Data can be transferred in either of two codes, extended binary coded decimal interchange code (EBCDIC) or the IBM version of the American National Standard Code for Information Interchange (American National Standards Institute, 3.4—1968; this code is called ASCII in this publication). The customer must specify which code he will use at the time he orders the BSCA, BSCC, or ICA. (Only units using the same code can communicate with each other.)

EBCDIC is the standard, 8-bit plus parity, internal binary code of the IBM System/3. (This code is illustrated in Appendix A.) The parity bit, used for internal checking, is not transmitted over the communications network.

ASCII is a 7-bit code plus parity. It is illustrated in Appendix A. Unlike EBCDIC, which numbers its bits 0 through 7 starting at the high-order bit, ASCII numbers its bits 1 through 7 starting at the low-order bit (Figure 10-1).

All characters are transmitted over the line low-order bit first. For ASCII, the high-order bit must be a 0-bit from main storage on transmit. If the adapter does not receive a high-order 0 from main storage, it generates and transmits a wrong parity (P) bit. In addition, the invalid ASCII character status bit is set on causing a unit check condition.

On receive, the first bit received is transferred into low-order main storage position and so on. For ASCII, the adapter fills a 0 into the high-order bit position in main storage except when the character has a VRC error.

EBCDIC and ASCII have different coding structures to represent characters. When ASCII is used with a System/3 communications adapter, the program must translate data from EBCDIC before transmission and to EBCDIC after reception.

	First hex				Second hex			
	High				Low			
Transmission	8	7	6	5	4	3	2	1
EBCDIC	0	1	2	3	4	5	6	7
ASCII	P	7	6	5	4	3	2	1
Autocall dial digit (BCD)[1]	X	X	X	X	8	4	2	1

[1] Applies to BSCA only.

Figure 10-1. Bit Positions and Significance

STANDARD SUBFEATURES OF THE BSCA AND ICA

Two subfeatures of the BSCA and ICA are standard: intermediate block checking and auto-answer. The auto-answer feature (switched network only) enables the communications adapter to respond to a telephone request for data communications automatically without operator intervention if the data set has unattended answer capability. The intermediate block checking feature allows transmission and reception of checking characters for checking the accuracy of communication without interrupting the steady flow of information from the transmitting station to the receiving station.

STANDARD SUBFEATURES OF BSCC

The following subfeatures are standard with BSCC:

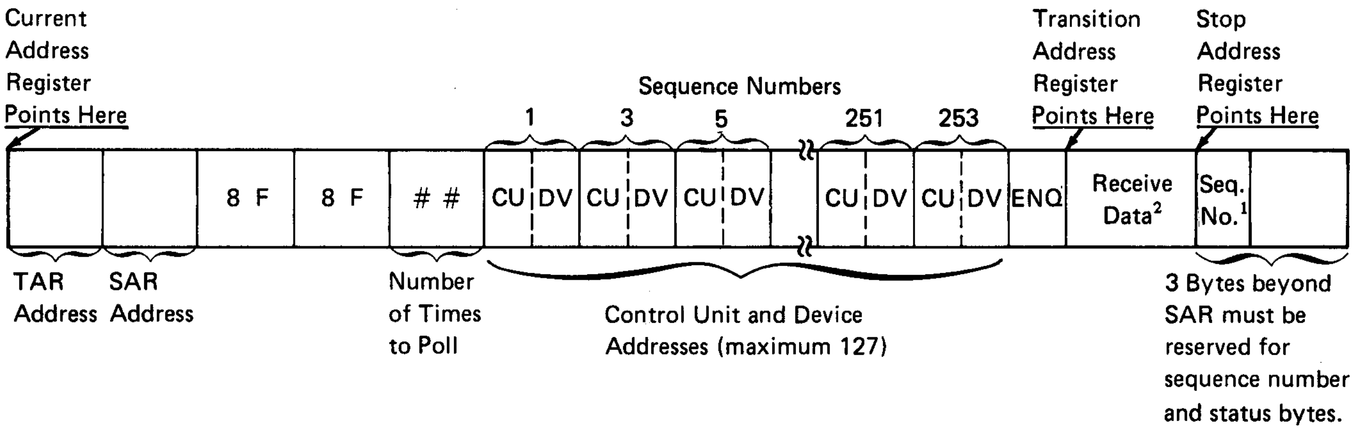
- Automatic polling of 3270 type terminals
- Transparency (with EBCDIC only)
- ASCII or EBCDIC transmission codes (see *Transmission Codes* in this chapter)
- Nonswitched, multipoint control station (see *Network Configuration* in this chapter)
- Half-duplex operation (see *Network Configuration* in this chapter)

Automatic Polling by BSCC

The automatic polling function is invoked with a normal transmit/receive instruction but the data field in the CPU must be organized as shown in Figure 10-2.

When the command to execute a transmit/receive instruction is received, the BSCC begins to cycle steal data from the CPU at the address indicated by the current address register. The first 4 bytes of the data field are the address values of the transition address register and the stop address register respectively. The next 2 bytes are examined to see if they contain hex 8F8F. If they do contain a hex 8F8F, then BSCC does not transmit these bytes and goes into an automatic polling mode. The next byte contains the number of times (any hex value from 01 through FE) the BSCC is designated to go through its polling list before it discontinues automatic polling. If this byte contains hex FF, the BSCC goes into a continuous polling mode. The remaining bytes contain the control unit and device address (CUDV) up to a maximum of 127 terminals.

Control unit and device addresses can be entered in a string for any desired polling sequence. When the addresses of all terminals are entered, ENQ is inserted to close the string. When the ENQ character is detected, automatic polling commences.



[1] This is the sequence number of the active CUDV at the time of status.
[2] If the receive data does not fill the buffer area, the sequence number and status bytes can be positioned ahead of where the SAR is pointing. However, 3 bytes beyond SAR should always be reserved.

Figure 10-2. BSCC Data Field Format for Automatic Polling Instruction

Automatic polling continues without interrupts until one of the following conditions occur:

- A terminal is polled four times without responding.
- A terminal responds to the polling with data.
- All the terminals are polled the designated number of times.
- An SIO command is received to stop the polling.

Under these conditions, an op-end interrupt is generated and the sequence number and status bytes are posted. If all the terminals are polled the designated number of times, an EOT is also placed in the data field immediately following the ENQ.

The sequence number is always an odd number from 1 to 253 (1, 3, 5, 7, etc) that indicates the CUDV in the string in use at the time the status was posted. The status bytes and their meanings are shown in Figure 10-3.

Status Byte 2 (hex)	Status Byte 1 (hex)	Meaning
00	02	'Data set ready' on, no micro detected errors
00	03	Main storage data overrun during auto poll caused by CAR equals SAR during transfer of auto poll buffer to the system
00	0A	'Data terminal ready' not on after an SIO command
00	10	'Data set ready' not on after SIO command
00	1A	Invalid N-code for BSCC SIO command
00	2A	Invalid buffer service request condition, both transmit and receive bits on
00	32	Invalid buffer service request condition, data link escape 2 on and cycle steal byte 1 off
00	3A	Invalid buffer service request condition, both transmit and receive bits off
00	42	Invalid transmit state, CAR equals TAR but no change of direction or intermediate text block character received from the system
00	4A	Invalid state of cycle steal byte 1 and 2 buffers during transmit
00	52	Invalid state of cycle steal byte 1 and 2 buffers during receive
00	5A	Invalid receive condition, CAR equals SAR but no intermediate text block character received from the line
00	6A	Timeout during SIO transmit setup, waiting for clear to send
00	72	Invalid auto poll message format, missing second hex 8F
00	7A	256-byte auto poll buffer full and no end-of-transmission block or end of text received from the line
00	82	256-byte auto poll buffer full and no ENQ character received from the system

Figure 10-3 (Part 1 of 3). Auto Poll Status Bytes

Status Byte 2 (hex)	Status Byte 1 (hex)	Meaning
00	9A	BSCC line error, 'data terminal ready' not on during transmit setup
00	A8	BSCC line error, 'data set ready' not on during transmit setup
00	B2	BSCC line error, 'request to send' not on during transmit setup
00	BA	BSCC attachment error, transmit mode not on
00	C2	Invalid entry flag bit on, both LIO and SIO flags are off
00	CA	Invalid I/O instruction: – LIO CAR (N = 4) issued with line already busy – SIO issued with SIO already in progress – SIO issued without issuing LIO CAR first
00	D2	Invalid IR byte for SIO microcontroller command, N-code equals 5
00	E2	Invalid IR byte for SIO diagnostic command, N-code equals 6
00	EA	Invalid control word in auto poll routine: – SIO auto poll issued with the number of times to poll equal to 00 – Microcode control word invalid in auto poll routine – Microcode address pointer invalid in auto poll routine
00	FA	Received ASCII VRC error (wrong parity from line)
X1	YY	Line wrap test failure, YY defined in above errors (X can be any value)
08	02	Invalid ASCII character received from the system
10	02	Adapter check during receive, hardware error caused overrun
10	62	Adapter check during receive, timeout on store Cycle Steal request to the system
10	A2	Adapter check during receive, microcode caused overrun
20	22	Adapter check in transmit, timeout on fetch Cycle Steal request to the system
20	F2	Timeout during transmit
40	02	Received block check character or longitudinal redundancy check character data check, data from line is bad
00	DA	Received data check, end-of-transmission block or end of text received without start of text
40	FA	Received ASCII longitudinal redundancy check and vertical redundancy check error (wrong parity from line)

Figure 10-3 (Part 2 of 3). Auto Poll Status Bytes

Status Byte 2 (hex)	Status Byte 1 (hex)	Meaning
80	A2	Received timeout during auto poll
80	F2	Timeout during receive, but not auto poll

Figure 10-3 (Part 3 of 3). Auto Poll Status Bytes

Figure 10-2 shows the status bytes and sequence bytes located in the field after SAR. In the event the data area is not filled, these bytes could fall into an address within the stop address. In either event, the status bytes are always the two bytes preceding the value of CAR at op-end and the sequence byte is the third byte preceding CAR at op-end.

Another capability of automatic polling is automatic retransmission of the BSCC buffers. This occurs when a terminal, having data to transmit, responds to polling. The data (up to 256 bytes) is transmitted to the BSCC buffer and then to main storage. If, during the transfer of this data from the BSCC buffer to main storage, the allocated data area is exceeded, an interrupt occurs.

This condition is displayed in the status bytes and causes a NAK to be transmitted. The BSCC intercepts this NAK and instead of transmitting it to the terminal, the BSCC retransmits the data contained in its buffer to main storage.

Full Transparent Text Mode

This feature allows all of the 256 possible bit combinations available in EBCDIC to be transmitted through the communications adapter as data. This feature is necessary because certain of the EBCDIC characters are designated as line control characters and cause the communications adapter to perform a function. The transparency feature allows these control characters to be handled as data.

Network Configuration

The BSCC is designed for leased line, two- or four-wire, half-duplex operation. It is only supported as a multipoint control station by the system software.

OPTIONAL SUBFEATURES OF THE ICA

Full Transparent Text Mode (Special Feature)

This feature allows all of the 256 possible bit combinations available in EBCDIC to be transmitted through the communications adapter as data. This feature is necessary because certain of the EBCDIC characters are designated as line control characters and cause the communications adapter to perform a function. The transparency feature allows these control characters to be handled as data. This feature excludes the ASCII option.

8000 bps Local Interface

This feature permits local attachment, without the use of a modem or communications line, of one IBM 3271 Control Unit (Model 1 or 2) or one IBM 3275 Display Station (Model 1 or 2). The external modem cable of the attached 3271 or 3275 connects directly to the processing unit. This feature provides clocking for a data transfer rate of 8,000 bps.

2400 bps Local Interface

This feature permits local attachment, without the use of a modem or communications line, of one binary synchronous IBM terminal such as the IBM 3741 (Model 2 or 4). The external modem cable of the attached terminal connects directly to the processing unit. This feature provides clocking for a data transfer rate of 2,400 bps.

Synchronous Line, Medium Speed

This feature provides one medium-speed, binary synchronous line interface to an external modem. The communications network attachment may be point-to-point (switched or nonswitched) or multipoint (control station). Maximum transmission rate is 4,800 bps for switched operation, 9,600 bps for nonswitched operation. The attached modem must provide the necessary data clocking.

OPTIONAL SUBFEATURES OF THE BSCA

Station Selection (Special Feature)

This feature allows the system to operate as a tributary station in a multipoint communications network. This feature excludes the Auto-call feature and is not available with the high-speed interface selective feature.

Internal Clock (Special Feature)

This feature provides an internal clocking system in the communication adapter to allow operation with data sets that do not provide clocking to the adapter. The Internal Clock feature provides the following transmission rates:

600	bits per second
1,200	bits per second
2,000	bits per second
2,400	bits per second

Only one of the above transmission rates can be specified for each communication adapter. (Stations can communicate only with other stations using the same transmission rate.) This feature excludes the High-speed Interface selective feature.

High-Speed Interface (No-Charge Selective Feature)

This feature (which is used only with the first BSCA) enables the communication adapter to interface with data sets that provide data rates between 19,200 bits per second and 50,000 bits per second. This feature excludes the Internal Clock feature, so the data set must furnish data clocking when this feature is installed.

1200 bps Integrated Modem (Special Feature)

This feature eliminates the need for a stand alone modem or data sets between either the first or second BSCA feature and telephone facilities. The feature lets the BSCA operate at 1,200 bits per second on either (1) a leased half-duplex or duplex network or (2) a switched network.

The 1200 bps Integrated Modem feature is housed in the BSCA feature within the processing unit. Data interchange with the communications facility is serially by bit, serially by byte using frequency-shift keying (FSK) modulation. Modem clocking is performed by the BSCA internal clock, which is a prerequisite feature.

The 1,200 bps Integrated Modem feature is available in two versions:

- The *nonswitched leased line version* attaches to a type 3002 line facility by means of a cable supplied for the Modem feature.
- The *switched line version* provides automatic answering as a standard function and attaches to a type CBS or equivalent common carrier facility by a cable supplied for the Modem feature. Neither version can be installed on a BSCA that is equipped with the Auto-call feature.

Note: If the 1200 bps Integrated Modem feature is installed in a BSCA that is equipped with the Rate Select feature (not available in USA) the 1200 bps Modem can operate at either 600 bits per second or 1,200 bits per second under switch control.

Auto-call (Special Feature)

This special feature establishes automatic connection with a remote station on a switched network by a program instruction. An automatic calling unit (ACU), not supplied by IBM, must be used with this feature to permit automatic connection. This feature excludes the station selection feature and cannot be installed on a BSCA equipped with a 1200 bps Integrated Modem feature.

Full Transparent Text Mode (Special Feature)

This feature allows all of the 256 possible bit combinations available in EBCDIC to be transmitted through the communications adapter as data. This feature is necessary because certain of the EBCDIC characters are designated as line control characters and cause the communications adapter to perform a function. The transparency feature allows these control characters to be handled as data. This feature excludes the ASCII option.

Rate Select Switch (Special Feature)

Systems installed outside the USA that use data sets capable of operating at two rates are equipped with rate select switches. The rate select switch allows the system to operate at either 600 bits per second or 1,200 bits per second, according to the switch setting selected.

EIA Local Attachment (Special Feature)

The EIA Local Attachment feature allows attachment of a BSCA device such as the IBM 3270 Information Display System (via an IBM 3271 Control Unit) or an IBM 3275 Display Station in the same local environment without adapting the data signals from either the BSCA or the attached device for network transmission. The local attachment feature is installed in the processing unit; it is equipped with a connector to which the signal cable from the 3271 is connected. The feature supplies clocking for both the BSCA and the 3271 at data rates of either 2,400, 4,800, or 8,000 bits per second, as specified for the installation.

The EIA Local Attachment feature excludes the Internal Clock special feature and the attachment of any data set or IBM line adapter to the BSCA housing the EIA feature.

OPTIONAL SUBFEATURES OF THE BSCC

In addition to the standard subfeatures available, certain optional subfeatures are offered to enhance the capabilities of the BSCC.

Second Line

A second BSC line is offered as an option with BSCC. This line can have any of the subfeatures available with BSCC. It need not be configured the same as the first line.

Internal Clock (Special Feature)

This feature provides an internal clocking system in the BSCC to allow operation with data sets that do not provide clocking to the adapter. The Internal Clock feature provides a 1,200 bits per second clock rate. A half rate of 600 bits per second can be obtained by using the rate select feature.

The following EIA line speeds are supported by BSCC:

- 1,200 bits per second
- 2,000 bits per second
- 2,400 bits per second
- 4,800 bits per second
- 7,200 bits per second
- 9,600 bits per second

Only one of the above can be specified for each communication line.

The external modem must provide the clocking unless the EIA local feature or the Internal Clock feature is installed.

Synchronous Line, Medium Speed—EIA (Special Feature)

This feature provides one medium-speed, binary synchronous line interface to an external modem. The communications network attachment must be multipoint (control station). Maximum transmission rate is 9,600 bps for non-switched operation. The attached modem must provide the necessary data clocking or the internal clock feature must be installed.

EIA Local Attachment (Special Feature)

The EIA Local Attachment feature allows attachment of a BSCC device such as the IBM 3270 Information Display System (via an IBM 3271 Control Unit) or an IBM 3275 Display Station in the same local environment without adapting the data signals from either the BSCC or the attached device for network transmission. The local attachment feature is installed in the processing unit; it is equipped with a connector to which the signal cable from the device is connected. Only one device can be attached per local line.

The EIA Local Attachment feature requires the Internal Clock special feature.

1200 bps Integrated Modem (Special Feature)

This feature eliminates the need for a standalone modem or data sets between BSCC and telephone facilities.

The 1200 bps Integrated Modem feature is housed in the BSCC feature within the processing unit. Data interchange with the communications facility is serially by bit, serially by byte using frequency-shift keying (FSK) modulation. Modem clocking is performed by the BSCC internal clock, which is a prerequisite feature.

The 1200 bps Integrated Modem feature operates at 1,200 bits per second over nonswitched (two- or four-wire) facilities. The feature attaches to a 3002-type channel by means of an IBM provided cable.

Note: If the 1200 bps Integrated Modem feature is installed in a BSCC that is equipped with the Rate Select feature, the 1200 bps Modem can operate at either 600 bits per second or 1,200 bits per second under switch control.

Data-Phone Digital Service Adapter (Special Feature)

The Data-Phone[1] Digital Service Adapter (DDSA) is an integrated adapter that attaches to the nonswitched Data-Phone Digital Service (DDS) network. The DDSA interfaces with a DDS Channel Service Unit (CSU). Line speeds of 2,400, 4,800, and 9,600 bps are available.

Connection to the CSU is via an external cable that must be ordered separately.

Rate Select Switch (Special Feature)

Systems that use data sets capable of operating at two rates are equipped with rate select switches. The rate select switch allows the system to operate at either full rate or half rate, according to the switch setting selected.

Request-to-Send Tie-up (Special Feature)

This feature eliminates the modem turnaround delay between request-to-send and clear-to-send on a four-wire nonswitched network.

New Sync Connection (Special Feature)

This feature permits the new sync signal to be inserted into the interface cable for those modems that require it.

[1]Trademark of the American Telephone & Telegraph Co.

Protective Ground to Frame Ground Strap (Special Feature)

This feature is provided for use in those World Trade countries that require the protective ground be tied to the frame ground.

TERMINALS SUPPORTED BY BSCC

The following terminals and systems can be connected to a BSCC line. All terminals must be equipped with the BSC feature. Unlike the display adapter, BSCC does not emulate any controller so all terminals must be connected to a control unit or be capable of connecting directly to a BSC line:

- 3276 Control Unit Display Station (up to 7 devices in 3270 compatible mode)
- 3274 Control Unit (up to 32 devices in 3270 compatible mode)
- 3271 Control Unit
- 3278 Display Station
- 3275-1, -2 Display Station
- 3277-1, -2 Display Station
- 3284-1, -2 Printer
- 3286-1, -2 Printer
- 3288 -2 Printer
- 3735
- 3741-2, -4 Data Station
- 5231-2
- 3600 system (via 3601/3602 with BSC RPQ)
- System/3
- System/7
- System/32

Terminals are controlled by software architecture similar to that which controls a 3277 (or 328x) attached to a 3271 attached to a System/3 BSCA.

BSCC PROGRAMMING

The BSCC and the processing unit interface through the I/O channel using System/3 instructions for communications.

Note: Due to the asynchronous microprocessor, an attempt to clock step through the BSCC instructions can result in a processor check.

LOCAL COMMUNICATIONS ADAPTER (LCA)

The Local Communications Adapter feature directly attaches (no data set/modem) an IBM 3741 Data Station Model 2, an IBM 3271 Control Unit, or an IBM 3275 Display Station to an IBM System/3. The LCA is installed in the processing unit. The external (data set/modem) cable furnished with the attached device (3741-2, 3271, or 3275) is plugged directly into a connector provided with the LCA feature.

Only one device may be physically attached to the LCA at a time. The LCA provides clocking at a rate of 2,400 bits per second for the attached device and operations in a point-to-point, nontransparent mode using extended binary coded decimal interchange code (EBCDIC).

The LCA cannot be installed on a system with an installed first BSCA, and only one LCA can be installed. However, a system can be equipped with an LCA and a second BSCA feature. None of the BSCA subfeatures can be used with the LCA feature.

Registers and programming required for operation of the first BSCA feature are used for the LCA feature.

DISPLAY ADAPTER AND LOCAL DISPLAY ADAPTER

The display adapter and the local display adapter perform the same functions and are controlled identically. However, they have the following differences:

- The display adapter (DA) can be installed on the System/3 Model 15, and a maximum of 30 terminals can be attached.
- The local display adapter (local DA) can be installed on the System/3 Model 12 or the System/3 Model 8, and a maximum of 12 terminals can be attached.

For the rest of this section, information that applies to the display adapter also applies to the local display adapter unless otherwise specified. The DA performs a control and I/O interfacing function with the terminals, and channel communication with the processing unit. DA communication with the processing unit is via the standard I/O channel. DA communication with the terminals is by means of a single coaxial cable to each terminal, with a maximum cable length of 2,000 feet per cable.

Terminals that can attach to the DA are the 3277, 3284, 3286, and 3288 Models 1 and 2.

Terminals are controlled by software architecture similar to that which controls a 3277 (or 328x) attached to a 3271 attached to a System/3 BSCA.

The DA emulates BSCA/3271. Only BSCA EBCDIC point-to-point nonswitched support is provided. All 3271 features (including Katakana) are supported except ASCII and transparent monitor mode.

The DA and BSCA-2 are mutually exclusive. The DA uses the BSCA-2 channel address, interrupt level, and cycle steal priority.

Data transfer between the DA and the terminals is serially by bit (13 bits per word). Odd parity is contained within the word and is checked at the receiving end.

The following list summarizes the DA functions:

- Perform data serialization/deserialization and error detection functions.
- Provide in-transit message storage buffer and control on data transfer operations. Data is transferred between the DA and the terminals in groups of 480 or 1920 words (depending on the terminal model). Control and polling commands are on a single word basis.
- Transfer data between the DA buffer and the processing unit main storage on a cycle steal basis.
- Identify and respond to commands issued by the processing unit.
- Assemble terminal and attachment status for error recovery and diagnostic evaluation by the processing unit.

3277 CRT/Keyboard

The following list summarizes the 3277 CRT/keyboard functions:

Common 3277 functions:

- Contain serializer/deserializer logic, control, and status registers.
- Receive character codes and function codes from the keyboard. As each character is keyed, it is displayed on the CRT.
- Contain circuitry to check parity of received data and generate parity for transmitted data.
- Support a 64-character set.

3277 Model 1 functions:

- Display 480 characters (12 lines of 40 characters each).
- Contain a 480-byte message buffer and image generation circuitry.

3277 Model 2 functions:

- Display 1,920 characters (24 lines of 80 characters each).
- Contain a 1,920-byte message buffer and image generation circuitry.

3284, 3286, and 3288 Printers

The following list summarizes the printer functions:

Common printer functions of the 3284 and 3286:

- Use a matrix print head and pin feed platen.
- Print characters by a series of dots within a 7 x 7 matrix.
- Use a character set of 64 EBCDIC characters.
- Produce a maximum print line of 132 characters; however, shorter lines can be printed if the data is formatted by system programming.

Printer functions of the 3284 and 3286 characteristics of printer types and models:

Terminal	Model	Characters Printed/Second	Character Buffer Size
3284	1	40	480
3284	2	40	1920
3286	1	66	480
3286	2	66	1920

Printer functions of the 3288 line printer:

- Print 120 lines per minute.
- Has a character buffer size of 1,920 bits.
- Use a character set of 64 EBCDIC characters.
- Produce a maximum print line of 132 characters; however, shorter lines can be printed if data is formatted by system programming.

Continuous Poll by Display Adapter

The display adapter provides a continuous poll function that does not exist in the BSCA/3271. The program may use a device address of hex 8F followed by a count, a list of device addresses to be polled and the normal ending ENQ. The 8F must be repeated twice as if it were a normal device address. The count is a 1-byte hexadecimal number. The device address list contains 1 byte for each device address entry. It must contain valid device addresses and may be a maximum of 255 bytes long. A unit check results if invalid devices are specified or if the list contains greater than 255 entries.

Upon receipt of a continuous poll, the display adapter polls the specified devices in the order given (devices may be repeated more than once to set up priorities, if desired). Assuming that there is no I/O pending or status pending in any device in the list, the adapter restarts the list each time that the last entry is polled. Each time the list has been polled 100 times, the adapter subtracts one from the list. When the count goes to 0, polling stops and the normal interrupt and EOT response occur. The display adapter is busy and the processing unit usage meter runs while a continuous poll is in progress.

If at any time during polling I/O pending or pending status other than device busy is found in any device in the list, polling ends and the normal response and interrupt occurs. Device busy status is ignored.

The count may be any hex value between 01 and FF. A 0 count must not be used. Any count between hex 01 and FE is treated as described above. However, if a count of hex FF is specified, the continuous poll does not end until an I/O pending or pending status is found in one of the devices or until the program issues a start 2-second timeout instruction. When the continuous poll is stopped by a start 2-second timeout instruction, no 2-second timeout occurs, and the adapter initiates a normal interrupt and EOT response to indicate that polling has stopped.

DISPLAY ADAPTER AND LOCAL DISPLAY ADAPTER PROGRAMMING

The processing unit and DA interface through the I/O channel using two types of instructions: attachment and BSCA-2.

Adapter instruction use channel address 4 with M-bit of 1, or channel address 5 with M-bit of 1. They are used to:

- Initialize the attachment
- Enable the attachment and microcontroller
- Detect errors
- Recover from errors (reinitialization)

BSCA-2 instructions use channel address 8 with M-bit of 1. They are used to:

- Control terminals
- Control BSCA-2 (DA) interrupts
- Enable/disable BSCA-2 (DA)

The BSCA-2 instruction provide the processing unit with a means of controlling terminals using the existing BSCA/3271/3277/3284/3286/3288 software.

Initializing the Display Adapter and Local Display Adapter

The display adapter uses a microcontroller to perform many functions under control of a microprogram provided by IBM. This microprogram must *never* be altered. Before operating the adapter after a power down condition and after an attachment check condition, (1) adapter must be initialized; (2) the microprogram must be loaded into control (microcontroller) storage; then (3) the attachment and microcontroller must be enabled.

If you are using IBM programming support, all of these functions are performed during the IPL procedure. Otherwise, loading CE deck FFF, then CE deck 893, then CE deck FC7 initializes the adapter, loads the microprogram, and enables both the adapter and the microcontroller.

Initializing the Display Adapter and Local Display Adapter without IBM Programming or CE Decks

Following power on, the program must issue the following sequence of attachment commands:

1. 8 Attachment LIOs (any data) to the HDBs to insure proper parity in the 16 low-order bytes. (LIOs and SNS commands to the HDB before the attachment is enabled are directed to the low 16 bytes of the HDB). SNS I/O commands must not be issued until the op decode registers have been loaded.
2. 32 Attachment LIOs to the op decode to provide the op decode with the proper information.
3. An attachment SIO to enable the attachment.
4. 8 Attachment LIOs (any data) to the HDBs to insure proper parity in the next 16 bytes. (LIOs and SNS commands to the HDB after the attachment is enabled are directed to the 2nd 16 bytes of the HDB.)
5. A sequence of 32 attachment LIOs, each one loading one of the op decode registers.
6. A sequence of 32 attachment SNS instructions, each sensing one of the 32 op decode registers.
7. The microcontroller may now be enabled.

LOCAL STORAGE REGISTERS USED BY COMMUNICATIONS FEATURES

Three local storage registers (two of which are either located or in the case of BSCC emulated in the adapter) are provided for the communications feature: the current address register, the transition address register, and the stop address register. These registers hold the storage addresses of data or line control characters at which certain actions are to occur, or the address of the next byte to be transmitted or received.

Current Address Register (CAR)

The current address register contains the address of the next byte to be operated on. When data is being transmitted, this register is used to address storage for each byte that is to be transmitted. When data is being received, this register is used to address storage for storing each byte as it is received from the line. The address is increased by 1 under control of the adapter during every I/O cycle steal.

Transition Address Register (TAR)

The transition address register stores the address at which a reversal is desired between transmitting and receiving in a transmit-and-receive operation. When the address in the current address register equals the address in the transition address register, the adapter stops taking data from storage on cycle steals and begins stealing I/O cycles to store the characters received from the communications line.

Stop Address Register (SAR)

The stop address register stores the address at which the communications adapter I/O operation must stop. When the address in the current address register equals the address in the stop address register, the communications adapter ends its operation and generates an interrupt request.

COMMUNICATIONS FEATURES CONTROL

Communications features controls are called into action at each station by:

- Starting codes, to enter certain modes and to begin to accumulate block check characters (BCC)
- Modifies, sync characters, and data link escape functions (ITB, SYN, DLE)
- Ending codes, to terminate blocks and activate checking functions

Control Characters and Sequences (Figure 10-4)

When transmitting, the adapter turns around to receive when the current address register is equal to the transition address register. The program must ensure that the last character of the change-of-direction sequence is at a location 1 less than the transition address. When receiving, any change-of-direction character or sequence causes the adapter to terminate the receive operation and issue an op-end interrupt request.

- SOH or STX resets control mode and sets the adapter to data mode. The first SOH or STX after line turnaround resets the BCC buffer, and BCC accumulation begins with the following character.
- ETB or ETX resets data mode in the adapter and is the last character included in the BCC accumulation. At the master station, the adapter transmits the BCC and the pad character. At the slave station, the adapter compares its BCC accumulation with the BCC received following the ETB or ETX.
- For recognition of EOT or NAK as a control character, the adapter requires that 4 contiguous 1-bits must be received immediately following the EOT or NAK. Also, the EOT character must be the first non-SYN character after establishing character sync. On transmit, the adapter automatically generates the 4 contiguous 1-bits by sending the trailing pad character.
- ENQ resets data mode in the adapter.

- SYN is generated and transmitted automatically by the adapter to establish and maintain synchronism. SYN does not enter BCC or main storage. A SYN from main storage at the transmitting station is transmitted, but does not enter main storage at the receiving station nor BCC accumulation at either station.

- SYN SYN is the sync pattern in nontransparent mode. Two contiguous SYN characters are always transmitted immediately following an ITB or XITB, BCC sequence. SYN is also used as a time fill character for a transmit only instruction terminated by ITB or XITB until the next transmit and receive instruction is issued.

- ITB is included in the BCC and causes the BCC(s) to be sent or received. Both adapters continue in data mode with the new BCC accumulation starting with the first non-SYN character.

- DLE alerts the adapter to test the following character for a defined control sequence. In nontransparent data mode, DLE is treated as data.

- XSTX resets control state and sets the adapter to data mode and transparent mode. Unless preceded by SOH ——, XSTX resets the BCC register and BCC accumulation begins with the following character. In transparent mode, the first DLE in each two-character DLE sequence does not enter BCC or main storage. The second character does, if it is not SYN. Also, the transmitting adapter inserts an extra DLE for each DLE received from main storage.

- XSYN is the sync pattern for maintaining synchronism in transparent mode. It does not enter BCC or main storage.

- XENQ resets data mode and transparent mode in the adapter.

- XETB or XETX causes the same adapter action as ETB or ETX and, in addition, resets transparent mode.

- XITB causes the same adapter action as ITB and, in addition, resets transparent mode.

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	ITB	IUS	US
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[1]	TTD	STX ENQ	STX ENQ
Transparent start of text	XSTX	DLE STX	
Transparent intermediate block	XITB	DLE IUS	
Transparent end of text[1]	XETX	DLE ETX	
Transparent end of trans. block[1]	XETB	DLE ETB	
Transparent synchronous idle	XSYN	DLE SYN	
Transparent block cancel[1]	XENQ	DLE ENQ	
Transparent TTD[1]	XTTD	DLE STX DLE ENQ	
Data DLE in transparent mode	XDLE	DLE DLE	

[1]Change of direction character.

Figure 10-4. Control Characters and Sequences

Pad Characters

The adapter generates and sends one pad character for each change-of-direction character transmitted. If the change-of-direction sequence calls for a BCC character, the pad character follows the BCC character; otherwise, the pad character follows the change of direction character in the message being transmitted. This pad character is hex FF.

The adapter also generates and transmits a hex FF (pad character) as the second character of the NAK and EOT control character sequences.

When transmission starts, the adapter automatically generates and inserts a pad character (in this case, a hex 55) ahead of the initial synchronizing sequence. No leading or trailing pad character (except a pad character immediately following either EOT or NAK) is stored during receive operations.

Adapter Synchronization

The adapter receives timing pulses externally from the modem which, in this case, establishes and maintains bit synchronism. The adapter starting to transmit automatically sends two SYNs required for establishing character synchronism at the receiving adapter. The receiving adapter establishes character synchronism by decoding two consecutive SYNs.

An adapter with the Internal Clock feature or EIA Local Attachment feature establishes and maintains bit synchronism on its own. For this purpose, the adapter automatically sends two additional hex 55 characters preceding the character synchronism pattern.

To maintain character synchronism, the transmitting adapter (master) inserts a synchronization pattern, SYN SYN, at every transmit timeout. The synchronization pattern does not enter BCC or main storage. In transparent mode, the transparent synchronous idle is used.

If a transmit only operation is terminated with ITB or XITB, the synchronization pattern, SYN SYN, is transmitted immediately following the BCC.

FRAMING THE MESSAGE, COMMUNICATIONS FEATURES

The program at the transmitting station must frame the data to be sent with appropriate line control characters. These characters are stored at the receiving station, so the program must allow space for them in storage. When transmitting, the adapter automatically generates and transmits SYN, pad and BCC (or LRC/VRC for ASCII) characters as required for establishing and maintaining synchronism with the remote station and for error checking. When receiving, the BSCA removes all SYN and BCC (or LRC/VRC) characters and some pad characters from the data before storing. The pad character following a NAK or EOT *is not* removed by the adapter.

Response characters (ACK0, ACK1, WACK, and NAK) are inserted by the stored program, not the transmitting adapter. They are not stripped by the receiving adapter. The program must store these characters in a known location so that the program can test them to determine what action to take next.

INTERRUPTS, COMMUNICATIONS FEATURES (Except BSCC)

The adapter initiates two types of level 2 interrupts: operation end (op end) interrupts and intermediate text block (ITB) interrupts. Whenever an interrupt occurs, the program must determine, by TIO ITB interrupt and TIO op-end interrupt instruction, the type of interrupt that occurred and which adapter is affected. The ITB interrupt latch and the op-end interrupt latch are reset by their respective TIO instructions; both latches are reset by disable adapter.

The interrupt pending condition, which is set by either the op-end or ITB interrupt latch, is remembered until it is reset by an SIO reset interrupt request instruction. When interrupts are disabled, the interrupt latches operate as when enabled, except that interrupt pending does not signal an interrupt request to the processing unit.

When two adapters are installed on System/3, determine which adapter is originating the interrupt request by issuing a TIO interrupt pending instruction (Figure 10-5). Interrupt pending indicates that either an ITB interrupt or an op-end interrupt is needed by the tested adapter. After determining which adapter caused the interrupt, the program issues appropriate TIO op-end and TIO ITB instructions, using the appropriate M-bit to specify the adapter requesting the interrupt.

Interrupt requests from any adapter except the BSCC should be serviced by routines similar to the one shown in Figure 10-5. Note that both types of interrupts must be tested and the ITB interrupt must be tested first.

Op-End Interrupt

Any attachment that uses line control logic (this includes BSCA, LCA, ICA, and DA) can present op-end requests to the system in which it is installed. All System/3 models can accept op-end requests from these attachments.

If enabled, an op-end interrupt occurs at the end of the following adapter operations:

- Auto-call (BSCA)
- Transmit and receive
- Receive initial
- Receive
- Loop test
- Two-second timeout (the adapter need not be enabled to complete the 2-second timeout operation with an op-end interrupt)

For auto-call, an op-end interrupt occurs after the connection is established or the call is abandoned.

In a receive type operation, an op end interrupt is generated when a change-of-direction character is decoded, when the current address equals the stop address, or when a receive timeout occurs.

In a transmit operation, the interrupt is generated when the current address, transition address, and stop address are all equal. In addition, if an adapter check occurs on transmit, the operation is immediately terminated and an op-end interrupt is generated.

In a loop test diagnostic operation, an op-end interrupt is generated when the current address is equal to the stop address.

On a start 2-second timeout operation, an op-end interrupt is generated at the end of the 2-second period.

ITB Interrupt on BSCA, LCA, and ICA

An ITB interrupt occurs at a slave station whenever interrupt is enabled, an ITB character is received, and no errors are detected.

The ITB interrupt should be serviced prior to the request for the next succeeding interrupt. (This period of time is a function of bits per second and number of bytes in the next intermediate block.) Allow time for processing unit interference caused by I/O cycle steals and by the need to service higher priority interrupts.

If the ITB interrupt is not serviced before the adapter receives the next ITB character, the next ITB interrupt request may be lost.

(The ITB interrupt is not processed by the display adapters.)

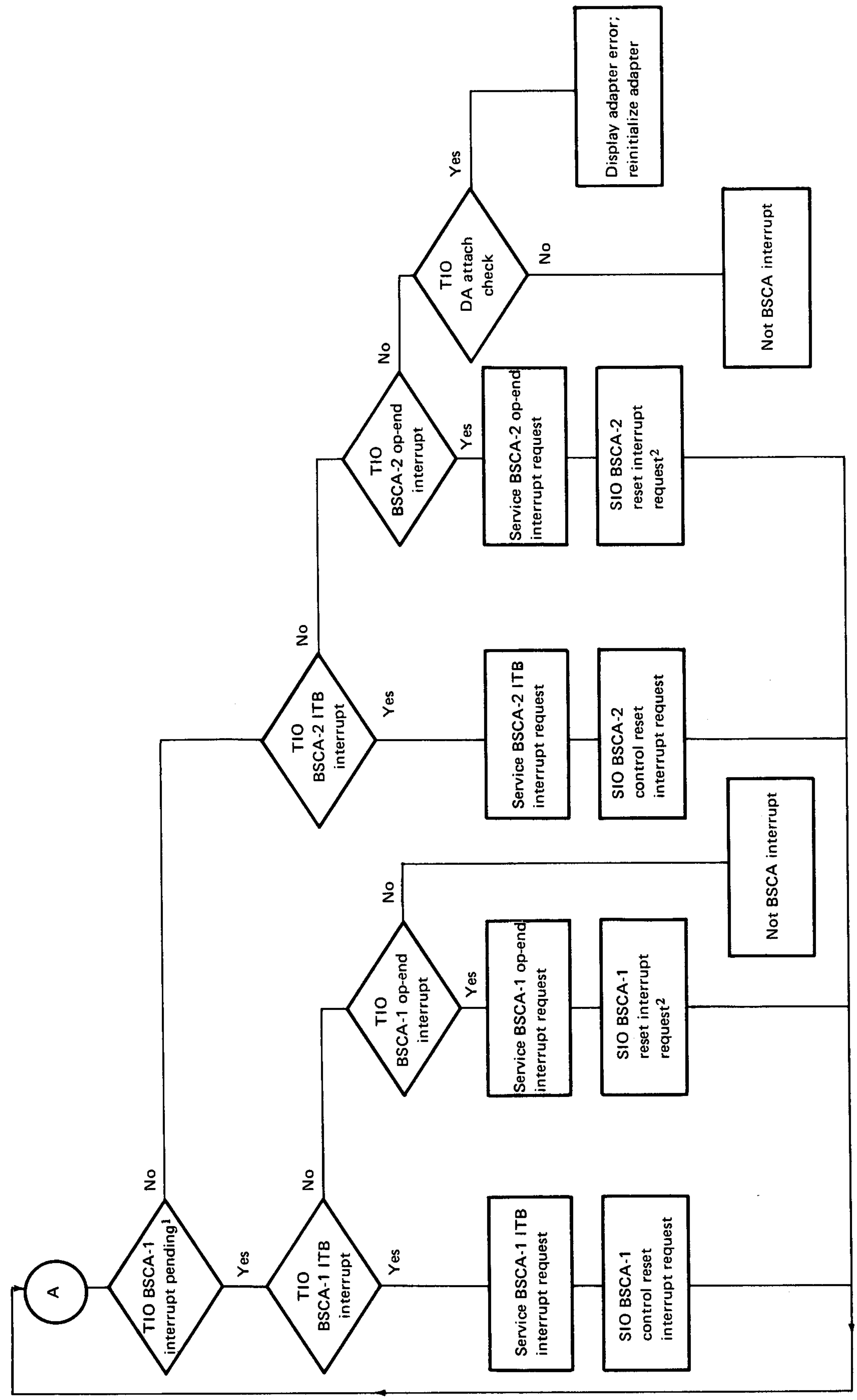


Figure 10-5. Generalized Communications Adapter Interrupt

[1]This instruction is not needed if the second BSCA feature is not installed.
[2]An op-end interrupt is normally reset and the next I/O operation, if any, is started by the last SIO instruction of the interrupt routine.

Note: Above references to BSCA-1 also apply to LCA, and references to BSCA-2 also apply to ICA & DA.

INTERRUPTS, BSCC

The BSCC initiates a level 3 interrupt for operation-end (op-end). Whenever the interrupt occurs, the program must determine by the TIO op-end interrupt instruction that the interrupt was a BSCC op-end. The op-end interrupt latch is reset by the TIO instruction.

The interrupt pending condition is set by the following:

- Op-end interrupt
- IMPL sequence complete
- I/O check
- No-op
- Microcontroller in wait state and IMPL sequence complete

Interrupt pending is remembered until it is reset by an SIO reset interrupt request instruction. When interrupts are disabled, the interrupt latch operates as when enabled, except that interrupt pending does not signal an interrupt request to the processing unit.

When two lines are installed on BSCC, to determine which line is originating the interrupt request, a TIO op-end instruction is issued with one of the lines selected and then with the other line selected. After determining which line caused the interrupt, the program issues appropriate instructions, using the appropriate line select to handle the request.

All BSCC interrupt requests should be serviced by routines similar to the one shown in Figure 10-6.

Op-End Interrupt

If enabled, an op-end interrupt occurs at the end of the following BSCC operations:

- Transmit and receive
- Receive initial
- Receive

In a receive type operation, an op-end interrupt is generated when a change-of-direction character is decoded, when the current address equals the stop address, or when a receive timeout occurs.

In a transmit operation, the interrupt is generated when the current address, transition address, and stop address are all equal. In addition, if an adapter check occurs on transmit, the operation is immediately terminated and an op-end interrupt is generated.

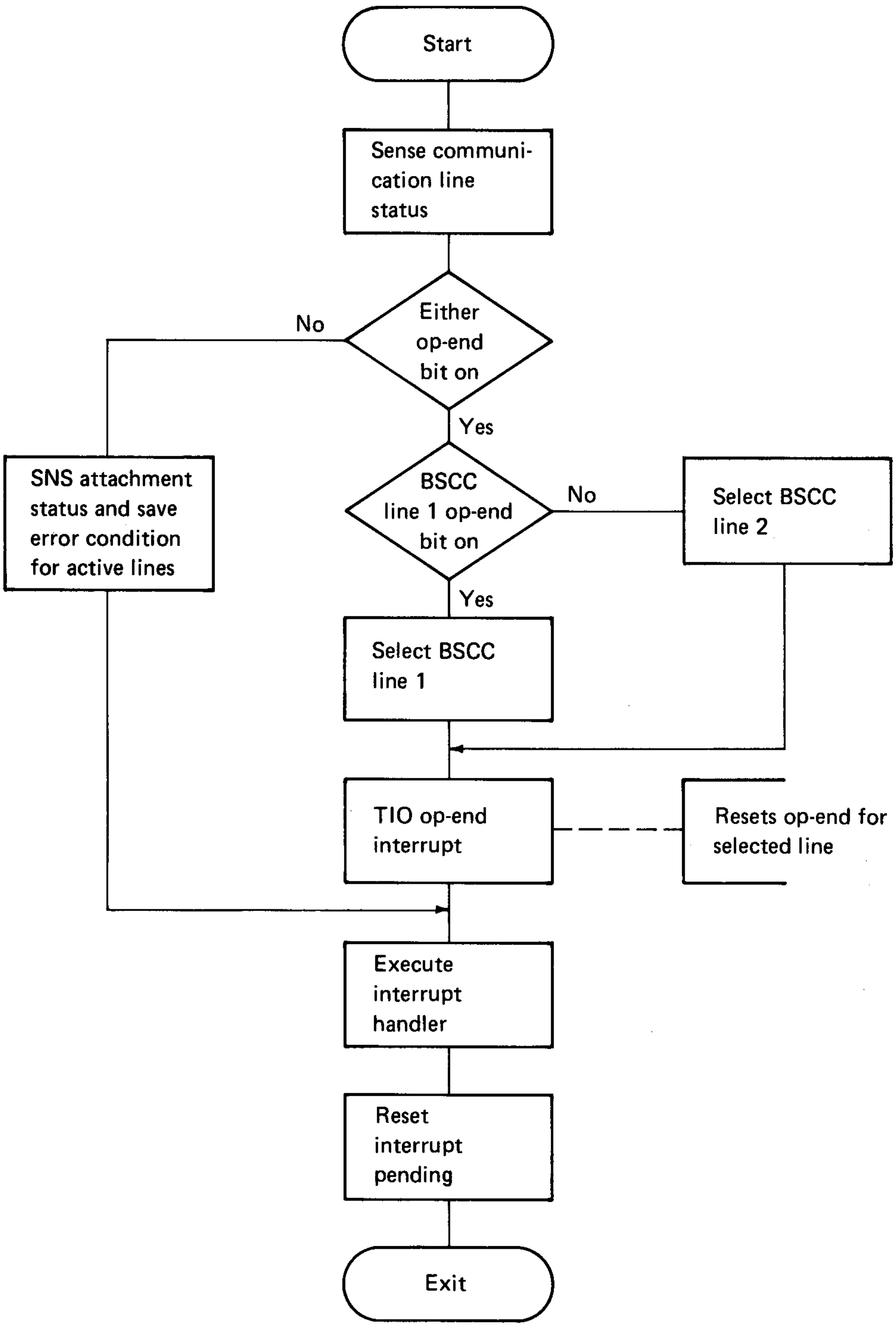


Figure 10-6. Generalized BSCC Interrupt

BSCA/LCA/ICA/DA START I/O (SIO)

Op Code (hex)	Q-Byte (binary)	R-Byte (binary)
Byte 1	Byte 2	Byte 3
F3	1000 x xxx	xxxx xxxx

DA M N — Control Code

Bits

0123 4567 Function Specified

Bit 7:
- 0 = No function specified
- 1 = Reset interrupt request

Bit 6:
- 0 = Disable interrupt request
- 1 = Enable interrupt request

Bit 5:
- 0 = Cancel 2-second timeout
- 1 = Start 2-second timeout

Bit 4: Not used

Bit 3:
- 0 = Disable step mode[1,2]
- 1 = Enable step mode[1,2]

Bit 2:
- 0 = Disable test mode[1,2]
- 1 = Enable test mode[1,2]

Bit 1:
- 0 = Disable adapter[1]
- 1 = Enable adapter[1]

Bit 0:
- 0 = Disregard bits 1, 2, and 3
- 1 = Activate bits 1, 2, and 3 (see definition of bits 1, 2, and 3 for further definition)

N-Code Operation

N-Code	*Communications Adapters*	*Display Adapters*
000	Control	Control
001	Receive only	Invalid
010	Transmit and receive	Transmit and receive
011	Receive initial	Invalid
100	Auto-call	Invalid
110	Loop test	Invalid

An N-code not shown is invalid and causes:
- Program check if interrupt level 7 is enabled on Model 15
- Processor check if interrupt level 7 is not enabled on Model 15
- Processor check on Models 8, 10, and 12

M:
- 0 = Select BSCA-1/LCA
- 1 = Select BSCA-2/ICA/DA

DA: Hex 8 specifies the communications/display adapter as the device to be controlled.

F3 specifies a start I/O operation. Hex F as the first hex digit in the op code indicates that the instruction is a command (that is, no operand addressing is involved).

[1] If bit 0 = 0, this bit has no meaning.
[2] Bit has no meaning for display adapter.

Operation

The adapter specified by the M-code performs the actions specified by the N-code and control code.

Program Notes

- The start I/O instruction initiates all communications adapter operations. While the communications adapter is busy or is not-ready for any reason except unit check, the program will not accept any start I/O instruction except control. Issuing the start I/O when the communications adapter is in the not ready condition causes the I/O attention light and adapter attention light on the system control panel to light. Correcting the not ready condition causes the instruction to be executed.
- See display adapter attachment instructions in this section for information about controlling and testing the display adapter attachment.

Functions

Control: The N-code that specifies the control function provides only the functions specified by the control code. This is the only instruction that can initiate the 2-second timeout function.

Receive Only: This operation accepts characters from the line and places them in storage at the location designated by the current address register. The adapter updates the current address register plus one each time a character is stored. The receive only operation ends: (1) when a change of direction character is received from the line, (2) when the current address register equals the stop address register, or (3) when no synchronizing characters are received from the line for 3 seconds.

Any of the control functions except start 2-second timeout can be initiated by this instruction.

Transmit and Receive: This function takes characters from storage at the location designated by the current address register and transmits them on the line to the remote station. The adapter updates the current address register plus 1 as it transmits each character. The last character to be transmitted must be a change-of-direction character and must be stored at an address 1 less than the address contained in the transition address register.

When the current address register is updated to equal the transition address register, the communications adapter stops transmitting and begins receiving characters from the line, storing the characters received into main storage at locations specified by the current address register. The adapter updates the current address register plus 1 as it stores each character.

The operation ends and the adapter generates an interrupt request when: (1) a change-of-direction character is received, (2) the current address register equals the stop address register, or (3) no synchronizing characters are received for 3 seconds. Any of the control functions except start 2-second timeout can be initiated by this instruction.

The transmit-and-receive operation can be used as a transmit only operation (this is mandatory for transmitting transparent ITB blocks) by loading the same address into both the transition address register and the stop address register. A transmit-and-receive operation with a zero length transmit field (initial value of the current address register and transition address register the same) is not allowed.

The transmit-and-receive function is provided to reduce line turnaround time. The transmit-and-receive operation should be used in all transmit sequences that require a response.

Receive Initial: This operation allows the remote station to establish contact so it can transmit a message. The receive-initial function is the only one that can be used by a tributary station for establishing contact in a multipoint network. In this operation, the local communications adapter monitors the line until it receives an initialization sequence. Upon receiving the initialization sequence, the communications adapter stores the characters received in locations specified by the current address register. The adapter updates the address register by +1 as each character is stored. The operation ends and the adapter generates interrupt request when: (1) the adapter recognizes a change-of-direction character, (2) the current address register equals the stop address register, or (3) no synchronizing characters are received for 3 seconds after an initialization sequence is begun. Any of the control functions except start 2-second timeout can be combined with this operation.

Auto-call: This function is provided as a special feature in the communications adapter. In operation, the communications adapter takes the number to be called, one digit at a time, from storage locations specified by the current address reigster. Each digit to be dialed must be specified in BCD code in the digit portion of a byte. These numbers are sent by BSCA logic to an automatic calling unit (ACU) that dials the number of the remote station. The BSCA updates the current address register by +1 as each byte is transferred to the ACU. When the current address register equals the stop address register, the communications adapter stops sending digits to the automatic calling unit and waits for an indication of line connection having been established or of the call having been terminated. If the condition is established, the adapter is signaled to end the operation. If the call is terminated, the BSCA sets the timeout status bit, ends the operation, and generates an interrupt request. If the timeout status bit is on, the program should retry the operation after disabling the BSCA for 2 seconds.

Any of the control functions except start 2-second timeout or enable BSCA can be combined with this operation.

Loop Test: The loop test function is used by the customer engineer to test the functioning of the communications adapter. It is of no use to the problem programmer.

Reset Interrupt Request, Enable Interrupt, and Disable Interrupt Control: These functions control the communications adapter's ability to interrupt the main program. The adapter operates on interrupt level 2. Two kinds of interruptions can occur from the communications adapter: an ITB interruption and an operation-end (op-end) interruption. The interruption routine must determine with a test-I/O-and-branch instruction which type of interruption occurred. The ITB interruption should be serviced first.

The ITB interruption occurs during receiving operations when the adapter receives an ITB character if the block check characters indicate that everything transmitted in that block was received correctly. When the ITB interrupt occurs, the program can store the contents of the transition address register to indicate the point at which data in the next block begins in storage. All the data up to (but not including) this address is data to be processed. The status bytes cannot be sensed during an ITB interrupt because the bits in the status bytes apply to the data being received, rather than to the data that has been received (for ITB operation only).

Op end interruptions occur at the end of all the functions controlled by the N-code. In addition, the 2-second timeout causes an interruption 2 seconds after the CPU issues an SIO control instruction with a control code that specifies start 2-second timeout. Op-end interrupt routines usually sense the status byte to determine the status of the last operation. The status bytes are valid for op-end interrupts because no data is transferred between the interrupt request and the interrupt routine.

Because the communications adapter continues to receive data from the remote station during ITB interrupt routine servicing, the program should sense the transition address register before the next ITB character is received. The processing time available is a function of the data rate of the data set used and the number of bytes in the next intermediate block. Allow extra time in the interrupt routine to account for time that may be required for CPU interference caused by I/O cycle steals and by the occurrence of higher priority interrupts.

Two-Second Timeout: This SIO control code function is provided to obtain a 2-second delay before the transmission of TTD or WACK. The start 2-second timeout must be given only with the Q-code control function. When the timeout is completed, an interrupt is generated. The adapter is not busy when doing a 2-second timeout. It can be terminated by issuing any SIO with the control code specifying cancel 2-second timeout. A previously issued start 2-second timeout must be terminated if an SIO noncontrol instruction is issued. The start 2-second timeout operation mut not be issued while the adapter is busy.

The adapter need not be enabled to complete the 2-second timeout operation with an op-end interrupt.

Enable/Disable Step and Test Modes: These are diagnostic functions useful to the customer engineer but of no interest to the problem programmer.

Enable/Disable Adapter Control: The enable adapter function causes the communications adapter to become operable and allows it to connect to the data set and perform data handling functions. At this point, the program should issue a TIO not ready test instruction. The disable adapter function deconditions the adapter and disconnects it from the data set.

BSCA/LCA/ICA/DA LOAD I/O (LIO)

Op Code (hex)	Q-Byte (binary)	Operand Address	
Byte 1	**Byte 2**	**Byte 3**	**Byte 4**
31	1000 x xxx	Operand 1 address	
71	1000 x xxx	Op 1 disp from XR1	
B1	1000 x xxx	Op 1 disp from XR2	

DA M N

N-Code	Register to be Loaded
001	Stop address register
010	Transition address register
100	Current address register
110	Current address buffer (for diagnostic procedures only; should not be in user's program)

Any N-code not shown is invalid and causes:
- Program check if interrupt level 7 is enabled on Model 15
- Processor check if interrupt level 7 is not enabled on Model 15
- Processor check on Models 8, 10, and 12

M:
0 = select BSCA 1/LCA
1 = select BSCA 2/ICA/DA

DA: Hex 8 specifies the communications/display adapter as the device whose registers are to be loaded.

31, 71, or B1 specifies a load I/O operation. The first hex character in the op code specifies the type of operand addressing to be used for the instruction.

Operation

The CPU places the contents of the 2-byte field specified by the operand address into the register specified by the M-code and the N-code.

Program Note

If the program issues a load I/O instruction to the communications adapter while the adapter is busy, the adapter does not accept the instruction until the busy condition no longer exists.

BSCA/LCA/ICA/DA TEST I/O AND BRANCH (TIO)

Op Code (hex)	Q-Byte (binary)	Operand Address	
Byte 1	Byte 2	Byte 3	Byte 4
C1	1000 x xxx	Operand 1 address	
D1	1000 x xxx	Op 1 disp from XR1	
E1	1000 x xxx	Op 1 disp from XR2	

DA M N

N-Code	Condition Tested
000	Not ready/unit check
001	Op end interrupt
010	Busy
011	ITB interrupt
100	Interrupt pending
110	New data (diagnostic only) on communications adapter; invalid N-code for display adapters

Any N-code not shown is invalid and causes:
- Program check if interrupt level 7 is enabled on Model 15
- Processor check if interrupt level 7 is not enabled on Model 15
- Processor check on Models 8, 10, and 12

M: 0 = Select BSCA-1/LCA
1 = Select BSCA-2/ICA/DA

DA: Hex 8 specifies the communications features as the device to be tested.

C1, D1, or E1 specifies a test I/O and branch operation. The first hex character in the op code specifies the type of operand addressing to be used for the instruction.

Operation

The processing unit tests the adapter specified by the M-code for the condition specified by the N-code. If the tested condition exists, the processing unit branches to the instruction stored at the operand address. If the tested condition does not exist, the processing unit accesses the next sequential instruction.

Program Notes

- Whenever the processing unit detects that a tested condition exists, the processing unit places the address of the next sequential instruction in the address recall register and the branch-to address (from the operand address portion of the instruction) in the instruction address register. Then the processing unit accesses the instruction at the address stored in the IAR.
- Not-ready means (1) data terminal ready off, (2) ACU power off, (3) external test switch on and test mode disabled, or (4) data set ready latch off (nonswitched or multipoint network).
- The communications adapter becomes busy under different conditions, depending upon the kind of operation that is being performed. For all operations except receive initial, the adapter becomes busy as soon as the start I/O instruction is accepted; it remains busy until the operation ends. For receive initial operations, the following conditions cause busy.

  1. In a point-to-point nonswitched network, the adapter becomes busy as soon as the adapter establishes character synchronization with the remote station.
  2. In a point-to-point switched network, the adapter becomes busy as soon as the data set indicates that it received a call.
  3. In a multipoint network, the adapter becomes busy when it recognizes its own address in control mode.
- Unit check usually means that one of the status bits in status byte 2 is on.

BSCA/LCA/ICA/DA ADVANCE PROGRAM LEVEL (APL)

Op Code (hex)	Q-Byte (binary)	R-Byte (binary)
Byte 1	Byte 2	Byte 3
F1	1000 x xxx	0000 0000

R-Byte: This byte is not used for an APL instruction.

Q-Byte fields: DA, M, N

N:

N-Code	Condition Tested
000	Not-ready/unit check
001	Op-end interrupt
010	Busy
011	ITB interrupt
100	Interrupt pending
101	Invalid
110	New data (diagnostic only) on communications adapters; invalid N-code for display adapters

Any N-code not shown is invalid and causes:
- Program check if interrupt level 7 is enabled on Model 15
- Processor check if interrupt level 7 is not enabled on Model 15
- Processor check on Models 8, 10, and 12

M:
0 = Select BSCA-1/LCA
1 = Select BSCA-2/ICA/DA

DA: Hex 8 specifies the communications/display adapter as the device to be tested.

F1 specifies an advance program level operation. F, as the first hex character in the op code, specifies a command-type instruction (that is, an instruction without operand addressing).

Operation

This instruction tests for the conditions specified in the Q-byte.

- Condition present:
  - Systems with Dual Program Feature installed and enabled, activate the inactive program level.
  - Systems without Dual Program Feature installed or with Dual Program Feature installed but not enabled, loop on the advance program level instruction until the condition no longer exists.

- Condition not present: Systems with or without dual program feature access the next sequential instruction in the active program level.

Program Note

For additional information concerning the advance program level instruction, see Chapter 2.

BSCA/LCA/ICA/DA SENSE I/O (SNS)

Op Code (hex)	Q-Byte (binary)	Operand Address	
Byte 1	Byte 2	Byte 3	Byte 4
30	1000 x xxx	Operand 1 address	
70	1000 x xxx	Op 1 disp from XR1	
B0	1000 x xxx	Op 1 disp from XR2	

DA M N

N-Code	Register or Status Data
000	CE diagnostic
001	Stop address register
010	Transition address register
011	Status byte 1 and 2
100	Current address register
101	Invalid on communications adapters; CE diagnostic on display adapters
110	CE diagnostic
111	Invalid on communications adapters; CE diagnostic on display adapters

Any invalid N-code causes:
- Program check if interrupt level 7 is enabled on Model 15
- Processor check if interrupt level 7 is not enabled on Model 15
- Processor check on Models 8, 10, and 12

M: 0 = Select BSCA-1/LCA
1 = Select BSCA-2/ICA/DA

DA: Hex 8 specifies a communications/display adapter as the device to be sensed.

30, 70, or B0 specifies a sense I/O operation. The first hex character in the op code signifies the type of operand addressing to be used for the instruction.

Operation

The processing unit stores the contents of the register or status byte specified by the M-code and N-code in the 2-byte field specified by the operand address.

Program Notes

- The status bytes are bit significant as illustrated in Figure 10-7. Byte 1 is stored in the storage location addressed by the operand address; byte 2 is stored in the next lower storage location.

  The timeout bit is turned on by either of two conditions:

  1. Character synchronization is not established within 3.25 seconds from the start of a receiving operation.
  2. An automatic call operation is terminated by an abandon-call-and-retry signal from the automatic calling unit. This indicates that the call was not answered.

- Any noncontrol start I/O instruction resets the timeout bit.

- In a switched network, the disconnect-timeout status bit turns on if no heading, text, response, or control transmission occurs from either station for 20 seconds. A start I/O disable adapter instruction resets this bit. The 20-second disconnect-timeout function can be disabled by the customer engineer at installation time at the customer's request.

- The data-set-ready condition status bit is set on when the data set ready signal is detected and latched on. The bit is turned off if data set ready comes on and then turns off or if the communications adapter is not enabled.

- The data-line-occupied status bit turns on when the automatic calling unit signals that the data line is occupied, When this bit is on, a start I/O auto-call instruction or start I/O receive initial instruction is not accepted until the line is unoccupied. No start I/O auto-call or receive initial instruction should be issued in an interrupt routine when this bit is on.
- When the disconnect-timeout bit is on, the adapter automatically performed a disconnect operation.
- When a sense I/O transition address register or sense I/O stop address register instruction is executed, an adapter check can occur, causing a unit check indication. If this happens, it is possible that none of the byte 2 status bits are on.

Byte	Bit	Name	Indicates	Reset By
1 1 1 1 1 1	0 1 2 3 4 5	Not assigned		
1	6	Data set ready	This indicates that the data set is ready to operate and that the adapter is enabled. For adapters with local attachment feature, this indicates that the locally attached device is ready.	Data set going not-ready (For adapters with local attachment feature, the attached device is not-ready or the adapter is disabled.)
1	7	Data line occupied	(This bit is used on a switched network when the BSCA is equipped with the auto-call feature.) This bit indicates that the data line is busy and that any SIO auto-call or SIO receive initial instruction will be rejected. These instructions should not be issued during an interrupt routine with the data line occupied.	Data line becoming not busy
2	0	Timeout status	(1) A receive timeout occurred during a receive operation with the adapter in the busy state. (2) An auto-call operation was terminated by an abandon call and retry signal from the ACU (automatic calling unit), indicating that a connection was not established.	Any noncontrol SIO
2	1	Data check during receive operation	(1) A BCC compare check occurred (EBCDIC). (2) A VRC check occurred (ASCII). *Note:* Characters having VRC checks are distinguished by a high-order bit in main storage. These characters are never recognized as control characters by the adapter.	Any noncontrol SIO
2	2	Adapter check during transmit operation	(1) DBI register parity check; (2) I/O cycle steal overrun; (3) LSR or shift register parity check; (4) Transmit control register check. Adapter check on transmit terminates the operation and causes an immediate op-end interrupt.	Any noncontrol SIO
2	3	Adapter check during receive operation	(1) DBI register parity check; (2) I/O cycle steal overrun; (3) LSR or shift register parity check. Adapter check on receive does not terminate the operation.	Any noncontrol SIO
2	4	Invalid ASCII character	A byte fetched from main storage by an adapter using ASCII code contained a 1-bit in the high-order bit position.	Any noncontrol SIO

Figure 10-7 (Part 1 of 2). Status Indications

Byte	Bit	Name	Indicates	Reset By
2	5	Abortive disconnect	Indicates adapter switched network was enabled, then the data set became ready, then not-ready. This indicates the connection was released and causes data terminal ready to turn off. The program must allow enough time for a forced disconnect (adapter controlled) to occur. The program can use the 2-second timeout to ensure this.	SIO disable adapter
2	6	Disconnect timeout	Indicates disconnect timeout occurred on a switched network. Disconnect timeout causes data terminal ready to turn off. (May not apply to systems using the IBM remote job entry program.) *Note:* The program must perform a disconnect operation.	SIO disable adapter
2	7	Not assigned		

Note: When a SNS transition or SNS stop register instruction is executed, it is possible for an LSR, S-register, or DBI register parity check to occur. This can result in a unit check. Under this condition, the byte 2 status bits may all be 0.

Figure 10-7 (Part 2 of 2). Status Indications

BSCC ATTACHMENT INSTRUCTIONS

This special set of instructions are used to load, control, sense, and test the BSCC only.

BSCC START I/O (SIO)

Op Code (hex)	Q-Byte (binary)	R-Byte (binary)
Byte 1	Byte 2	Byte 3
F3	0010 0 xxx	xxxx xxxx

DA M N — Control Code

Control Code: Is active when N-code is 000 or 101

N-Code	Operation
000	Control

Bits 0123 4567	Function Specified
0000 0001	Enables interrupt request
0000 0010	Load micro-to-System/3 buffer
0000 0100	Not used
0000 1000	Set IMPL
0001 0000	Enable single cycle
0010 0000	Set micro reset
0100 0000	Enable Attachment
1000 0001	Disable interrupt request
1000 0010	Reset interrupt pending
1000 0100	Not used
1000 1000	Micro start clock pulse
1001 0000	Disable single cycle
1010 0000	Reset micro reset
1100 0000	Disable attachment

N-Code	Operation
001	Receive only
010	Transmit and receive
011	Receive initial
101	Microcontroller control

Bits 0123 4567	Function Specified
0000 0001	Start 2 second timer
0000 0010	Not used
0000 0100	Not used
0000 1000	Not used
0001 0000	Not used
0010 0000	Not used
0100 0000	Set test mode on
1000 0001	Cancel 2 second timer
1000 0010	Disable line selected
1000 0100	Not used
1000 1000	Not used
1001 0000	Not used
1010 0000	Stop polling
1100 0000	Set test mode off

N-Code	Operation
110	CE diagnostic

Any N-code not shown is invalid and causes a processor check.

M: Must be 0.

DA: Hex 2 specifies a BSCC as the device to be controlled.

F3 specifies a start I/O operation. F, as the first hex character in the op code, indicates that the instruction is a command (that is, no operand addressing is involved).

Operation

The BSCC performs the actions specified by the N-code and control code.

Program Notes

- The start I/O instruction initiates all BSCC operations. While the BSCC is busy, the program will not accept any start I/O instruction except control. Issuing the start I/O when the BSCC is in the not ready condition causes the instruction to be nonoperational. As a result, interrupt pending will be set. When the not ready condition has been corrected, interrupt pending can be reset and the SIO instruction reissued.
- There are three types of BSCC start I/O instructions. They are: (1) start I/O control to logic, (2) start I/O to microcontroller, and (3) functional start I/O.

Functions

Control: N-code 0 specifies control information for the interface logic. Specific actions to be performed are included in the I-R bits of the instruction. The I-R bits are only effective when N-code 0 is specified. An N-code 0 will be unconditionally accepted by the BSCC.

N-code 5 specifies control information for the BSCC microprocessor.

Receive Only: This operation accepts characters from the line and places them in storage at the location designated by the current address register. The BSCC updates the current address register plus one each time a character is stored. The receive only operation ends: (1) when a change of direction character is received from the line, (2) when the current address register equals the stop address register, or (3) when no synchronizing characters are received from the line for 3 seconds.

Transmit and Receive: This function takes characters from storage at the location designated by the current address register and transmits them on the line to the remote station. The BSCC updates the current address register plus 1 as it transmits each character. The last character to be transmitted must be a change-of-direction character and must be stored at an address 1 less than the address contained in the transition address register.

When the current address register is updated to equal the transition address register, the BSCC stops transmitting and begins receiving characters from the line, storing the characters received into main storage at locations specified by the current address register. The adapter updates the current address register plus 1 as it stores each character.

The operation ends and the BSCC generates an interrupt request when: (1) a change-of-direction character is received, (2) the current address register equals the stop address register, or (3) no synchronizing characters are received for 3 seconds.

The transmit-and-receive operation can be used as a transmit only operation (this is mandatory for transmitting transparent ITB blocks) by loading the same address into both the transition address register and the stop address register. A transmit-and-receive operation with a zero length transmit field (initial value of the current address register and transition address register the same) is not allowed.

The transmit-and-receive function is provided to reduce line turnaround time. The transmit-and-receive operation should be used in all transmit sequences that require a response.

Receive Initial: This operation allows the remote station to establish contact so it can transmit a message. In this operation, the BSCC monitors the line until it receives an initialization sequence. Upon receiving the initialization sequence, the communications adapter stores the characters received in locations specified by the current address register. The adapter updates the address register by +1 as each character is stored. The operation ends and the adapter generates interrupt request when: (1) the adapter recognizes a change-of-direction character, (2) the current address register equals the stop address register, or (3) no synchronizing characters are received for 3 seconds after an initialization sequence is begun.

Enable Interrupts: This function causes the BSCC to allow interrupt pending requests to generate an interrupt request to the processing unit.

Disable Interrupts: This function causes the BSCC to block any interrupt request from reaching the processing unit.

IMPL: This function causes BSCC to cycle steal data from main storage beginning at the IMPL start address and stopping at the IMPL stop address. The data is placed into the control store of BSCC. Interrupt pending is set when data transfer is complete. The attachments must be enabled and line 1 selected prior to issuing an IMPL instruction.

The microcode can be loaded in blocks if desired. The block need not be located contiguously in main storage, but data within the blocks must be contiguous.

The suggested IMPL sequence is as follows:

1. SIO to disable interrupts.
2. SIO to start microcontroller reset.
3. Load microcode into main storage.
4. LIO to place address of first byte of the block of microcode into the IMPL start address register.
5. LIO to place address of last byte of the block of microcode into the IMPL stop address register.
6. SIO to initiate IMPL sequence.
7. SIO to reset interrupt pending.
8. If interrupt is to be used to detect completion of the load of a block of microcode, issue SIO to enable interrupts. If this interrupt is not used, skip this step.
9. Interrupt pending is set when the loading of a block of microcode is complete. If interrupts are enabled, an interrupt to the system occurs. The IMPL block complete bit is also set at this time. This bit can be sensed.
10. If additional blocks of microcode are to be loaded, do the following:
    a. If interrupt is used to detect completion of a block load of microcode, issue an SIO to disable interrupt and an SIO to reset interrupt pending.
    b. If a TIO interrupt pending is used to detect completion of a block load of microcode, issue an SIO to reset interrupt pending.
    c. If an SNS of the IMPL block complete bit is used to detect completion of a block load of microcode, no action is required at this step.
11. Repeat steps 3 through 10 until all code is loaded.
12. If interrupts were enabled in step 8, issue an SIO to disable interrupts.
13. SIO to start microcontroller reset.
14. SIO to end microcontroller reset. Microcode begins executing at control store address 0000. This code tests internal microcontroller registers and microcode accuracy. Test results are loaded into the micro-to-System/3 buffer. The microcontroller then goes to a wait state. Successful test results are indicated by a completion code of hex 40. If the test is not successful, the entire IMPL procedure should be retried at least four times before aborting.
15. SIO to reset interrupt pending.
16. Test for successful microcode load by one of the following:
    - TIO interrupt pending (interrupts not enabled)
    - System/3 interrupt (interrupts enabled)
    - SNS attachment status and test for micro wait bit on
17. SNS micro-to-System/3 buffer. A successful test is indicated by the buffer containing a hex 40.
18. SIO to generate start clock pulse. This causes resumption of microcontroller execution.

Enable Attachment

This function allows the functions of the attachment to be active.

Disable Attachment

This function prevents the functions of the attachment to be active.

BSCC LOAD I/O (LIO)

Op Code (hex)	Q-Byte (binary)	Operand Address	
Byte 1	Byte 2	Byte 3	Byte 4
31	0010 0 xxx	Operand 1 address	
71	0010 0 xxx	Op 1 disp from XR1	
B1	0010 0 xxx	Op 1 disp from XR2	

DA M N

N-Code	Register to be Loaded
001	IMPL stop address register
010	Select line 1
011	Select line 2
100	Current address register or IMPL start address register
101	CE diagnostic

Any N-code not shown is invalid and causes a processor check.

M: Must be 0.

DA: Hex 2 specifies the BSCC as the device whose registers are to be loaded.

31, 71, or B1 specifies a load I/O operation. The first hex character in the op code specifies the type of operand addressing to be used for the instruction.

Operation

The CPU places the contents of the 2-byte field specified by the operand address into the register specified by the N-code.

Program Notes

- If the program issues a load I/O instruction to the BSCC while the BSCC is busy, the BSCC does not accept the instruction until the busy condition no longer exists.
- The operand address is not used for line select instructions (N-codes 2 and 3). The data transferred during these LIOs is ignored by BSCC.
- BSCC line 1 is also selected as a result of a system reset.
- When BSCC line 2 is selected, it remains selected until an LIO to select line 1 is executed or until a system reset.

BSCC TEST I/O (TIO)

Op Code (hex)	Q-Byte (binary)	Operand Address	
Byte 1	Byte 2	Byte 3	Byte 4
C1	0010 0 xxx	Operand 1 address	
D1	0010 0 xxx	Op 1 disp from XR1	
E1	0010 0 xxx	Op 1 disp from XR2	

DA M N

N-Code	Condition Tested
000	Not ready/unit check
001	Op end interrupt
010	System/3-to-micro buffer full
100	Interrupt pending
101	Micro-to-System/3 buffer full

Any N-code not shown is invalid and causes a processor check.

M: Must be 0.

DA: Hex 2 specifies a BSCC as the device to be tested.

C1, D1, or E1 specifies a test I/O operation. The first hex character in the op code specifies the type of operand addressing to be used for the instruction.

Operation

The processing unit tests the BSCC for the condition specified by the N-code. If the tested condition exists, the processing unit branches to the instruction stored at the operand address. If the tested condition does not exist, the processing unit accesses the next sequential instruction.

Program Notes

- Whenever the processing unit detects that a tested condition exists, the processing unit places the address of the next sequential instruction in the address recall register and the branch-to address (from the operand address portion of the instruction) in the instruction address register. Then the processing unit accesses the instruction at the address stored in the IAR.

- Not-ready means (1) BSCC is not enabled, (2) Microcontroller IMPL not complete, (3) I/O check condition, (4) I/O attention condition and that line selected, or (5) Microcontroller in a wait state and not in single cycle mode.

- Op-end interrupt indicates the BSCC has requested an interrupt because of the end of an operation. The TIO does not indicate which line caused the interrupt. This information is shown by the sense I/O communication lines status. The op-end interrupt for a specific line is reset by a test I/O instruction issued when the line causing the interrupt is selected.

- System/3-to-micro buffer full indicates that data in the buffer is ready to be serviced by the BSCC microcontroller. The System/3-to-micro buffer full bit is reset by the microcontroller after servicing.

- Interrupt pending indicates one of the following conditions occurred for the line selected:
  - IMPL block load complete
  - I/O check
  - No-op
  - Op-end
  - Microcontroller in a wait state

  When the interrupt pending is caused by the op-end (use test I/O op-end), all status required is available in the status bytes at the end of the data field.

  If the op-end did not cause the interrupt, it is recommended that the status bytes and sense I/O bytes be checked to determine the exact cause of the interrupt. Interrupt pending is reset by the sense I/O command to reset interrupt pending.

- Micro-to-System/3 buffer full indicates that the buffer is full and requests service from the system. The bit is reset by either a store I/O cycle or by a SNS micro-to-System/3 instruction.

BSCC ADVANCE PROGRAM LEVEL (APL)

Op Code (hex)	Q-Byte (binary)	R-Byte (binary)
Byte 1	Byte 2	Byte 3
F1	0010 0 xxx	0000 0000

DA M N — This byte is not used for an APL instruction

N:

N-Code	Condition Tested
000	Not ready/unit check
001	Op-end interrupt
010	System/3-to-micro buffer full
100	Interrupt pending
101	Micro-to-System/3 buffer full

Any N-code not shown is invalid and causes a processor check.

M: Must be 0.

DA: Hex 2 specifies a BSCC as the device to be tested.

F1 specifies an advance program level operation. F, as the first hex character in the op code, indicates that the instruction is a command (that is, no operand addressing is involved).

Operation

The APL instruction is identical to the test I/O instruction as far as the BSCC is concerned (that is, the same N-code, tested conditions, and responses apply).

BSCC SENSE I/O (SNS)

Op Code (hex)	Q-Byte (binary)	Operand Address	
Byte 1	**Byte 2**	**Byte 3**	**Byte 4**
30	0010 0 xxx	Operand 1 address	
70	0010 0 xxx	Op 1 disp from XR1	
B0	0010 0 xxx	Op 1 disp from XR2	

DA M N

N-Code	**Register or Status Data**
000	CE diagnostic
001	CE diagnostic
010	CE diagnostic and line 1 auto poll buffer
011	Status bytes 1 and 2
100	Current address register or IMPL start address register
110	Communication lines status and line 2 auto poll buffer

Low-order byte (operand address)

Bit	**Meaning**
0	BSCC line 1 busy
1	BSCC line 2 busy
2	Line 1 op-end interrupt
3	Line 2 op-end interrupt
4	BSCC line 1 select by System/3
5	BSCC line 2 select by System/3
6	Not used
7	Not used

High-order byte (operand address minus 1)

Bit	**Meaning**
0-7	Line 2 auto poll buffer

Any N-code not shown is invalid and causes a processor check.

Must be 0.

Hex 2 specifies a BSCC as the device to be sensed.

30, 70, or B0 specifies a sense I/O operation. The first hex character in the op code signifies the type of operand addressing to be used for the instruction.

Operation

The processing unit stores the contents of the register or status byte specified by the N-code in the 2-byte field specified by the operand address.

Program Notes

- The status bytes are bit significant as illustrated in Figure 10-8. Byte 1 is stored in the storage location addressed by the operand address; byte 2 is stored in the next lower storage location.

- The CAR-IMPL start address register contains the address of the storage location where the next byte will be placed. At op-end time the register contains the address one byte beyond the last status byte transferred by BSCC.

- The line 1 and line 2 auto poll buffers are not programmable.

- The contents of the line 1 auto poll buffer and the line 2 auto poll buffer is valid only when the microcontroller is stopped as a result of a micro detected error.

- During execution of the CE diagnostic instructions, the condition of the microdata bus in may be altered. Thus, concurrent operation of the micro during this time could cause indeterminate data on the micro data bus in.

Byte	Bit	Name	Indicates	Reset by
1	0	IMPL block complete	This indicates the IMPL block sequence was completed.	Power on reset or by issuing an SIO instruction to set the IMPL state.
1	1	Microcontroller error	Bad parity was detected at the control store output.	Reloading the microcode.
1	2	CE diagnostic		
1	3	CE diagnostic		
1	4	Microcontroller wait	The microcontroller is in a wait state.	Issuing an SIO instruction to generate a start clock pulse.
1	5	CE diagnostic		
1	6	CE diagnotic		
1	7	Not assigned		
2	0	No-op	The no-op bit is active. Once this bit is set, all succeeding functional SIO and LIO instructions will be handled as a no-op until this bit is reset.	A system reset, check reset, issuing an SIO instruction to reset interrupt pending, or a SNS instruction to sense attachment status.
2	1	Attachment not enabled	The attachment was not enabled with the appropriate SIO instruction.	Issuing an SIO instruction to enable the attachment.
2	2	Interrupts not enabled	The interrupts were not enabled with the appropriate SIO instruction.	Issuing an SIO instruction to enable the interrupts.
2	3	CE diagnostic		
2	4	I/O attention (line 1)	An I/O attention condition exists on line 1.	A system reset, check reset, or correcting the condition causing the I/O attention.
2	5	I/O attention (line 2)	An I/O attention condition exists on line 2.	A system reset, check reset, or correcting the condition causing the I/O attention.
2	6	CE diagnostic		
2	7	CE diagnostic		

Figure 10-8. BSCC Status Bytes

COMMUNICATIONS FEATURE OPERATIONS

Communications Feature Operations (Except BSCC)

The adapter controls all operations on the communication line through a combination of instructions in the System/3 processor and the automatic controls initiated by line control characters and sequences. Figure 10-9 is a basic flowchart of a suggested generalized routine to place the adapter in operation.

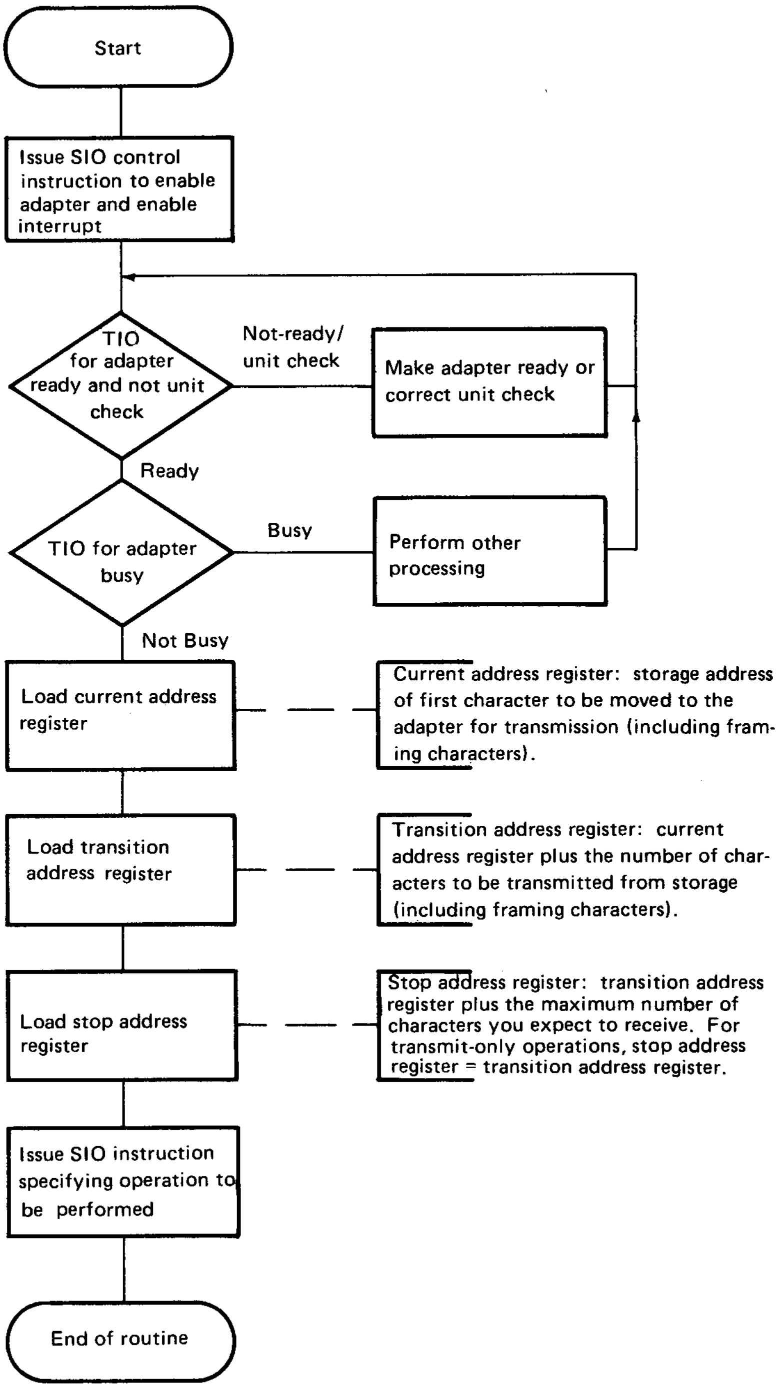


Figure 10-9. Initiating Communications Features Operation (Except BSCC)

BSCC Operations

The BSCC controls all operations on the communication line through a combination of instructions in the System/3 processor and the automatic controls initiated by line control characters and sequences. Figure 10-10 is a basic flowchart of a suggested generalized routine to place the BSCC in operation.

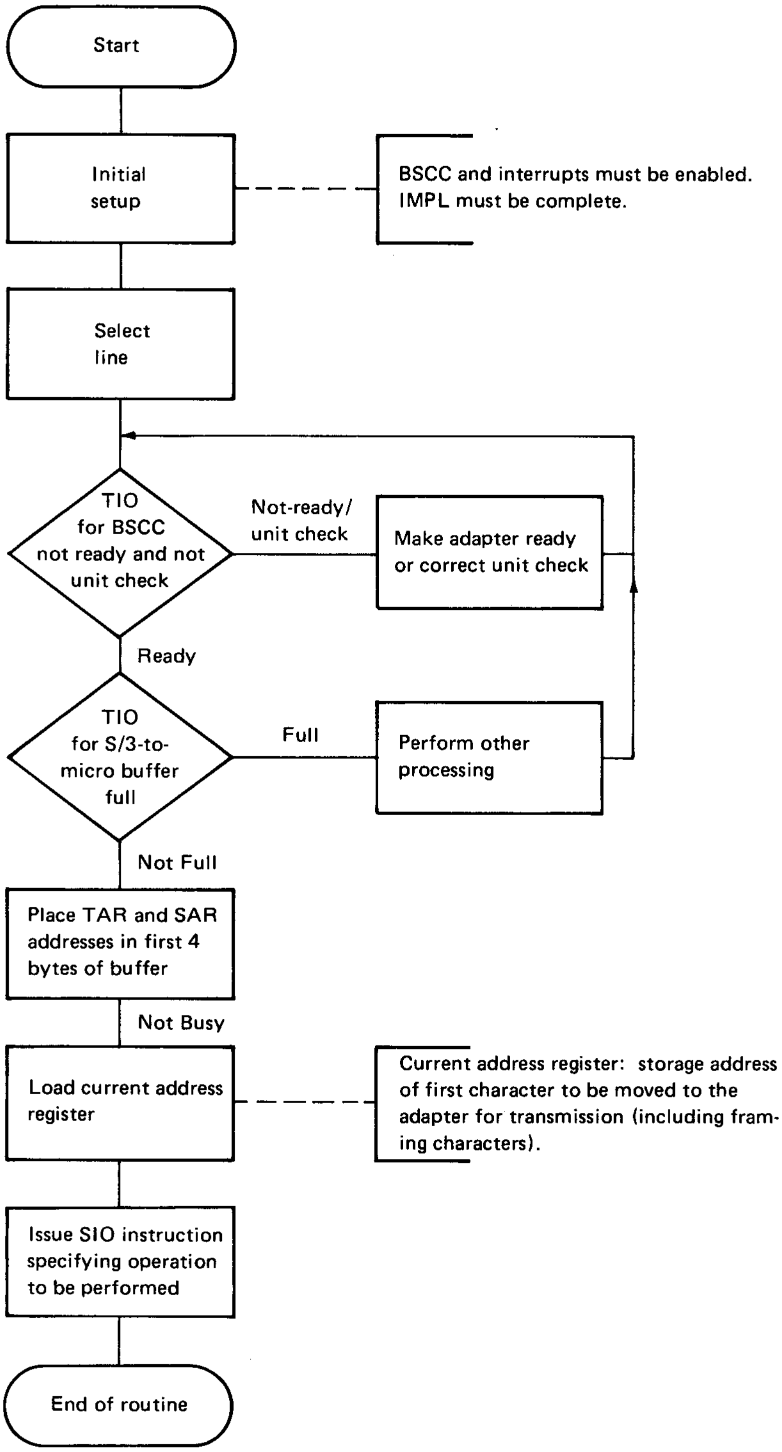


Figure 10-10. Initiating BSCC Action

Enable/Disable Communications Features (Except BSCC)

Enable adapter sets on the data terminal ready line to the data set; disable adapter sets off the data terminal ready line and resets the adapter. Power on reset or system reset or IPL also sets off the data terminal ready line and resets the adapter.

Since data terminal ready controls switching the data set to the communications channel, enable adapter is a prerequisite to establish a switched network connection. Disable adapter is used to disconnect from a switched network. Sufficient time must be allowed for the data set to disconnect from the switched network before the program again enables adapter. The 2-second timeout may be used to assure this.

Auto-call Operation—BSCA Only

At the calling station, data terminal ready must be on when the SIO auto-call instruction is issued. Auto-call should be issued as soon as possible after enable BSCA to avoid the possibility that another call comes in.

Prior to giving the auto-call instruction, the current address register and stop address register must be set up with LIOs to point to the number to be dialed. The stop address register must be set to the initial current address plus the number of digits to be dialed. The auto-call operation is executed by transferring bytes to the ACU at a data rate controlled by the ACU. Only the 4 low order bits in each byte from main storage are sent to the ACU. The transfer is on a cycle steal basis from the location specified by the current address register which is updated by +1 each cycle steal. This continues until the current address register is equal to the stop address register. At this point, the adapter waits for the ACU to signal that the connection was established or that the call was terminated.

An interrupt with no error condition indicates that a connection is established. If the timeout status bit is on (call terminated because of abandon call and retry signal from ACU), the program should retry the operation after disabling the BSCA for 2 seconds.

The SIO auto-call instruction is rejected and the I/O attention indicator set if the ACU power is off or data line occupied is on.

When the reject condition is removed by the operator, the SIO auto-call is accepted and the I/O attention indicator is reset.

Initialization Sequences

Initialization sequences are defined in *General Information Binary Synchronous Communications*, GA27-3004, and are transmitted by the transmit and receive instructions. Receive initial instruction is defined for receiving initial sequences. The receive initial operation depends on the data link (point-to-point nonswitched, point-to-point switched, or multipoint) selected by the customer.

Receive Initial Operation (Point-to-Point Nonswitched—Except BSCC)

On a nonswitched network, SIO receive initial causes the adapter to hunt for sync. When character sync is established, the adapter sets 'busy', 'receive timeout' then becomes effective, and the following sequence (starting with the first non-SYN character) is stored in the main storage area specified by the current address register. The stop address register should be loaded with the initial current address plus the maximum number of characters received. The operation is terminated and an interrupt generated when a change-of-direction character is received, the current address and stop address become equal, or a receive timeout occurs.

Receive Initial Operation (Point-to-Point Switched—Except BSCC)

On a switched network, SIO receive initial conditions the adapter to set 'busy' as soon as 'data set ready' comes up with the call. Receive timeout becomes effective and the adapter attempts to establish sync.

When character sync is established, the following sequence of received characters (starting with the first non-SYN character) is stored in the main storage area specified by the current address register. The stop address register should be loaded with the initial current address plus the maximum number of characters to be received. As above, the operation is terminated and an interrupt generated when a change-of-direction character is received, the current address and the stop address become equal, or a receive timeout occurs. In the case of a receive timeout, the recovery procedure is to issue the SIO receive only instruction.

Receive Initial Operation (Multipoint Tributary—BSCA and BSCC)

SIO receive initial is used to receive polling and selection sequences on a multipoint network. The stop address register should be loaded with the initial current address plus one less than the maximum number of characters in the polling/selection sequence. A two-character station address is used. For this operation, the low-order (rightmost) byte of the transition address register must be loaded with the station address. The EBCDIC 2-bit or the ASCII 6-bit of the first station address character received is disregarded; however, both characters of the address received must be identical.

For example, assuming EBCDIC code, if the transition address register is loaded with either XB or XS, the adapter recognizes either BB or SS as the station address. The high-order byte in the transition address register is not used.

The basic mode of BSCA/BSCC is in monitor mode for this operation. In this mode, the BSCA/BSCC hunts for sync. With character sync established, it monitors the line. All line control characters are decoded and the respective functions are executed, but data is not stored. When a valid EOT sequence is received, control mode is set.

In control mode, the BSCA/BSCC monitors for its station address. If it is not detected, the BSCA/BSCC continues monitoring the line. The adapter leaves control mode if no change-of-direction character is received within the receive timeout. A decoded SOH or STX drops control mode and puts the BSCA/BSCC back into monitor mode. If the station address is decoded as the first non-SYN characters after establishing character sync in control mode, the BSCA/BSCC immediately enters address mode, sets 'busy', and transfers the sequence starting with the second station address character, into the main storage area specified by the current address register. The operation is terminated and an interrupt is generated when a change-of-direction character is received, current address and stop address are equal, or when a receive timeout occurs.

BSCC uses this operation for diagnostic purposes only.

Auto-answer Wait Operation (Except BSCC)

The auto-answer wait function requires the following programming support: After adapter is enabled, an SIO receive initial instruction with interrupt enabled should be issued. The program can be stopped by a halt instruction; this stops the processing unit use meter. When the call is answered, 'busy' is set, causing the processing unit use meter to begin running. The op-end interrupt takes the processing unit out of the halt instruction to the adapter interrupt routine that must take the necessary programming action; for example, change the halt to a jump on condition, so that the mainline program starts when the interrupt routine is exited. The processing unit use meter continues running until normal job termination.

Transmit and Receive Operation

The SIO transmit and receive instruction is used for any type of transmission; that is, control sequences or text data. It sets the adapter to transmit mode, then takes characters from main storage and transmits them onto the line. BCC accumulation, data mode, and transparent mode are set depending on the type of line control characters fetched from storage. Transmission proceeds until the current address register equals the transition address register, which turns the adapter around to receive mode under the same instruction.

In receive mode, the adapter hunts for sync, then stores the characters received into main storage. As in transmit, the detail function on receive depends on the particular line control characters received.

The operation is terminated and an interrupt generated when an adapter check on transmit occurs, a change-of-direction sequence is received, the current address register equals the stop address register, or a receive timeout occurs. At this time, the unit check condition can be tested, and, if on, the status bits can be interrogated.

The reason for this combined transmit and receive instruction is the required fast response between the two operations. The effect of the current address, transition address, and stop addresses on the control sequences or text data is shown in Figure 10-11.

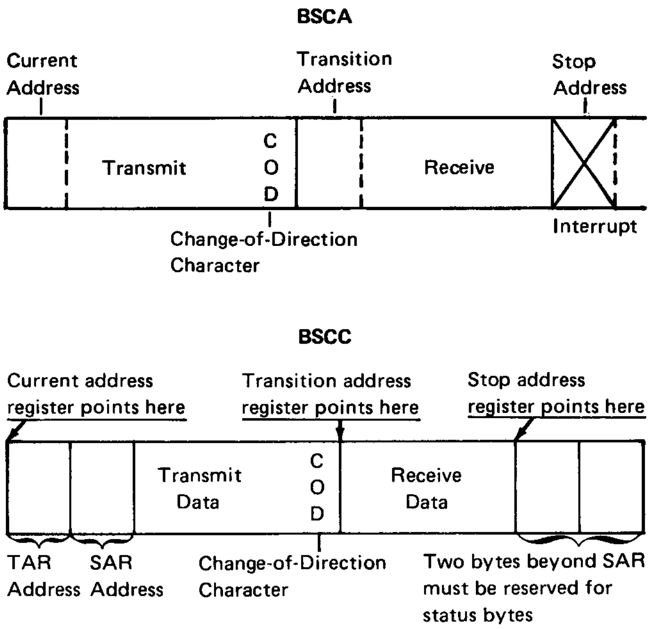


Figure 10-11. I/O Area and Address Register Contents at Start of Transmit and Receive Operation

The transmit and receive instruction is used by both the control and the tributary; that is, to send data and receive the reply, and to send the reply and receive data.

The current address specifies the beginning of the combined transmit-receive field and is updated by +1 on each cycle steal. The transition address register specifies the beginning of the receive field and must be loaded with the initial current address plus the number of characters to be transmitted. The stop address register specifies the end of the transmit and receive field and should be loaded with the transition address plus the number of characters to be received.

The current, transition, and stop addresses are 2-byte addresses that allow transfers of up to 64K bytes of data. A zero length transmit field is not permitted. If the stop address is equal to the transition address, the instruction becomes a transmit only operation.

At the start of the transmit and receive operation, the adapter sends one hex 55 character (two additional hex 55 characters if the Internal Clock Feature is installed) and two SYN characters. During transmit, the adapter inserts the sync pattern, SYN SYN, at every transmit timeout. SYN is not accumulated in the BCC and does not enter main storage. BCC compare takes place when an ITB, ETB, or ETX is received.

If the adapter entered data mode by receiving an STX or SOH, then only ETB, ETX, and ENQ are considered valid change-of-direction sequences. Outside of data mode, all turnaround sequences are considered valid change-of-direction sequences and will terminate the operation.

'Busy' stays on with the transmit and receive instruction throughout both sections of the operation until interrupt occurs. Interrupt occurs before the stop address is reached if a change-of-direction sequence is received.

ITB Operation

The IUS/US character is interpreted as the ITB control character to activate the ITB function. The control sends the BCC after the ITB, the tributary receives and compares it; both stations continue transferring more data immediately thereafter with no line turnaround.

For nontransparent data, the control can (1) transmit all ITB blocks in a single transmit and receive instruction or (2) transmit each ITB block in a transmit only instruction as described for transparent ITBs in the next section.

BSCA: When the slave receives an ITB character, the address plus 1 of where the character is in main storage is loaded in the transition address register. After the BCC comparison is made, and if no errors are detected, an ITB interrupt occurs. The adapter remains busy and proceeds to receive the next ITB block. The interrupt program, finding the 'ITB interrupt' latch on, stores the transition address register and processes the ITB block just received. Status bits are not sensed as they will apply to the subsequent block being received. Whenever a BCC error occurs, the adapter withholds the ITB interrupt for the ITB containing the error and for all the subsequent intermediate blocks, and stops sending data to storage. This continues until a change-of-direction character is recognized. When the ending sequence—ETB, ETX, or ENQ—is received, it is stored and an op-end interrupt occurs. At this time, the program checks the status bits to determine the appropriate reply.

BSCC: The ITB character is detected and the BCC checked, but no interrupt occurs. The BSCC remains busy and data transmission continues until the next COD character.

Transparent Operation

In transmitting and receiving data, transparent mode is set by the contiguous sequnce DLE STX. In transparency, the transmitting adapter automatically inserts a second DLE preceding each DLE from storage (except DLE STX), which is stripped by the receiving adapter. The additional DLE does not enter BCC accumulation.

Either ETB, ETX, ITB, or ENQ ends transparent mode at the master if it is at a location one less than the transition address. Due to this coincidence, the master adapter inserts a DLE so that the single DLE followed by ETB, ETX, ITB, or ENQ tells the slave to leave transparent mode. This DLE is stripped by the slave and is not included in the BCC at either station.

The use of the transition address to point at the control ETB, ETX, or ENQ allows replies to transparent data to consist of any number of characters. Limited conversational operation is possible in transparent, as well as nontransparent mode.

Each ITB block of transparent data must be transmitted with its own transmit and receive instruction. No turnaround takesplace after the ITB, and the adapter inserts at least two SYN characters (more, if necessary), until the next transmit and receive is issued or until 3 seconds elapse. During this period the adapter is not busy. Every ITB block must start out with DLE STX to again set transparent mode.

Disconnect Operation (Except BSCC)

The program can perform a disconnect operation on a switched network by giving an SIO disable adapter instruction, which drops the data terminal ready' line to the data set. It should previously transmit a DLE EOT sequence with a transmit and receive instruction to inform the other station that it is going *on-hook.* A received DLE EOT sequence should cause the slave station program to perform a disconnect operation.

If the 20-second disconnect timeout function has not been disabled, data terminal ready is also dropped by the disconnect timeout that occurs when there is no header, text, response, or control transmission on the line for 20 seconds.

Sufficient time must be allowed for the disconnect to occur before the program again enables adapter. The 2-second timeout may be used to assure this.

Receive Operation

The SIO receive instruction is defined for use when it is necessary to perform a receive operation after termination of the previous instruction, such as when a receive timeout has occurred. The operation is the same as the receive part of the transmit and receive operation. The adapter is busy for the entire operation.

This instruction must be used as a result of a receive timeout during a receive initial operation on a switched network.

Two-Second Timeout

This SIO control code function is provided to obtain a 2-second delay before transmitting a TTD or WACK. The start 2-second timeout must be given only with the N-code function control specification. When the timeout is completed, the adapter generates an interrupt. The adapter is not busy when doing a 2-second timeout. It can be terminated by issuing any SIO with the control code specifying cancel 2-second timeout. A previously issued start 2-second timeout must be terminated if an SIO noncontrol instruction is to be issued. Start 2-second timeout must not be issued if the adapter is busy.

The adapter needs to be enabled to perform the 2-second timeout operation.

Testing and Advancing Program Level

The TIO and APL instructions can be given at any time to test certain conditions. The following chart indicates these conditions:

Communications Features (Except BSCC)	BSCC
Not-ready/unit check	Not-ready
Busy	System/3-to-micro buffer full
ITB interrupt	Micro-to-System/3 buffer full
Op-end interrupt	Op-end interrupt
New data	

Not-ready (except BSCC) means: (1) data terminal ready off, (2) ACU power off, (3) external test switch on and test mode disabled, or (4) 'data set ready' latch off (nonswitched multipoint).

Not-ready (BSCC only) means: (1) attachment not enabled, (2) microcontroller IMPL not completed, (3) I/O check, (4) microcontroller in wait state, or (5) I/O attention with line selected.

Unit check (except BSCC) means that one of the status bits in byte 2 is on. When an SNS transition or SNS stop register instruction is executed, it is possible for an LSR, S-register, or DBI register parity check to occur resulting in a unit check condition. Under this condition, the byte 2 status bits may all be 0.

Busy (except BSCC) means the adapter is executing a: (1) receive initial, (2) transmit and receive, (3) auto-call, (4) receive, or (5) loop test (diagnostic) instruction.

Interrupt pending (except BSCC) means that either 'ITB interrupt' latch or 'op-end interrupt' latch is on. ITB interrupt and op-end interrupt are used to determine the type of interrupt that occurred and are reset off when tested by TIO.

Loading the Registers

LIO is used to load the current address register, IMPL start address register and the IMPL stop address register. BSCC also uses LIO to select line 1 or line 2.

Sensing

The BSCC uses SNS to store: (1) the current address register, (2) IMPL start address register, (3) status bytes, (4) communications lines status, (5) line 1 and line 2 auto poll buffers, and (6) diagnostic bits.

The other communication features use SNS to store: (1) the current address register, (2) transition address register, (3) stop address register, (4) diagnostic bits, (5) CRC/LRC buffer, and (6) status bits.

Data Checking

As the remote station transmits messages, it generates block check characters from the data bits transmitted. As these bits are received at the local communications adapter, the adapter generates a similar block check character from the data bits it receives. Each time the remote station transmits an ITB, ETB, or ETX character, it also transmits its block check characters. The local communications adapter compares these block check characters that it receives from the line with the block check characters that it generated from the data bits it received from the line. If the block check characters generated by the local communications adapter do not match the block check characters received from the line, the CRC/LRC/VRC status bit is set. While servicing the interrupt resulting from an ETB or ETX character, the program must sample the status bits and determine if the block check characters match each other.

If the interruption is the result of an ETB or ETX character, the result of the block check compare determines which response character should be sent. The positive acknowledgement characters alternate; ACK0 is transmitted in response to even-numbered blocks and ACK1 is transmitted in response to odd-numbered blocks. The program is responsible for transmitting the correct positive acknowledgement. The first block of text transmitted is always considered an odd-numbered block. If the wrong acknowledgement character is returned, the master station assumes that a block of data or heading was missed and initiates an error recovery procedure.

When block checking is initiated by ITB, the result of the block check compare is not transmitted immediately. Instead, if the block check compare is equal, the communications adapter continues to receive and store characters. If the block check is incorrect, no more data is stored, no more ITB interruptions are generated, and the VRC/LRC/CRC status bit is set on to indicate that a block check noncompare occurred. When the next ETB or ETX character is received, it is stored and an interrupting is generated. The status bits are sensed and tested to determine if all data was received correctly. An ENQ character also terminates the receive operation.

The lost data check is a program function. When the current address register equals the stop address register and a valid ending character is received, a lost data error is indicated.

Suggested Error Recovery Procedures

At the end of every transmit and/or receive operation, the program should test the adapter for a unit check. If a unit check is detected, the program should sense the adapter for status bytes. Test the status bits and perform the procedures for recovering from the error in the order given in Figure 10-12. The program must check for lost data and analyze the last two characters received to detect an abnormal response error.

System and Error Statistics

The user program should accumulate the following information for each adapter as a diagnostic aid. These counters should be logged to disk storage at close time (disk systems only).

Transmission Statistics

1. A count of data blocks transmitted successfully, as proven by the receipt of valid affirmative responses.

2. A count of data blocks that result in a negative response from the slave.

3. A count of invalid or no-response replies to transmitted data blocks and to following ENQ control characters.

4. A count of slave station terminations (EOT in lieu of normal response to text).

5. A count of adapter checks on transmit operations.

6. For System/3 multipoint control station applications, a count of transmissions and transmission errors for each terminal on the multipoint network.

Reception Statistics

1. A count of data blocks received correctly.

2. A count of data blocks received with BCC (or VRC) errors.

3. A count of ENQ characters received in message transfer state as a request from the master station to transmit the last response. ENQ as response to a transmitted WACK should not be included.

4. A count of master station forward terminations (TTD/NAK EOT sequences).

5. A count of adapter checks on receive operations.

Priority	Status Byte	Status Bit	Error Condition	Error Recovery Procedure (Recommended Program Action)
1	2	4	Invalid ASCII character	All cases – Action 1
2	2	5/6	Abortive disconnect or disconnect timeout (not used on BSCC)	All cases – Action 1
3	2	2	Adapter check on transmit	Control mode – Action 5 Slave – Action 4 Master – Action 3
	2	3	Adapter check on receive	Control mode – Action 5 Slave – Action 4 Master – Action 3
4	2	0	Timeout	Receive initial (switched) – Action 8 Auto-call or control mode – Action 5 Slave – Action 4 Master – Action 3
5	2	1	CRC/LRC/VRC	Control mode – Action 5 Slave – Action 2 Master – Action 3
	Program detected error[1]	Program detected error[1]	Lost data (CAR = SAR on receive)	
6	Program detected error[1]		Abnormal response	Control mode – Action 5 Slave: Absence of initial STX or terminal ETB/ETX – Action 4 Master: Improper ACK immediately preceded by timeout – Action 6 Master: Any response other than proper ACK or EOT – Action 7

Action Table	
1.	Permanent error – operator restart.
2.	Transmit and receive NAK – data n times when a control station.
3.	Transmit and receive ENQ – last response n times.
4.	Issue receive portion of previous operation n times.
5.	Polling or selection sequence – retry polling or selecting failing station L times after sending an EOT sequence to ensure control mode at the tributary stations. Other than polling or selecting sequence – retry last operation M times.
6.	Transmit and receive last text. This is an intermediate action within a recovery procedure; it is taken by the master each time it transmits text, times out on receive, transmits ENQ, and receives the improper ACK. A system hangup will not occur because of the limitation on Action 3.
7.	Transmit and receive ENQ once. If response is NAK, do Action 6 n times. If invalid response reoccurs, do Action 1.
8.	Issue SIO receive instruction.

The value L should be a minimum of 3.
The value M should be equal to or greater than N.
The value N should be a minimum of 7.

When L, M, or N is reached (permanent error) the program should terminate the job and tell the operator the nature of the error condition by some means (such as the halt identifier). Operator intervention is then required and the procedure is either to completely restart the job or to continue with the next job.

[1]The program provides lost data detection.

Note: A processor check stop causes a hard stop.

Figure 10-12. Communications Features Error Conditions and Recovery Procedures

DISPLAY ADAPTER ATTACHMENT INSTRUCTIONS

The display adapter has a special set of attachment instructions that are used to load, enable and disable, sense, and test the adapter and its attachment. These instructions are used with, rather than replace, communications adapter instructions.

ATTACHMENT START I/O (SIO)

(For use with display adapter only)

Op Code (hex)	Q-Byte (binary)	R-Byte (binary)
Byte 1	Byte 2	Byte 3
F3	0101 1 000	xxx0 0x00

DA M N — Control Code

Bits 3, 4, 6, 7 not used

Bit 0 = 0, and:

- Bit 5 = 0 — Reset diagnostic control
- Bit 5 = 1 — Set diagnostic control

Bit 0 = 1, and:

- Bit 1 = 0 — Disable attachment
- Bit 1 = 1 — Enable attachment
- Bit 2 = 0 — Disable microcontroller
- Bit 2 = 1 — Enable microcontroller

N: 000

Any N-code not shown is invalid and causes:

- Program check if interrupt level 7 is enabled on Model 15
- Processor check if interrupt level 7 is not enabled on Model 15
- Processor check on Models 8 and 12

M-code must be 1.

DA of hex 5 specifies the display adapter attachment.

Hex F3 specifies a start I/O operation. Hex F as the first digit in the op code indicates that the instruction is a command (that is, no operand addressing is involved).

Operation

This instruction allows the program to enable, disable, and reset the display adapter attachment.

Program Notes

- The display adapter attachment must be enabled after power is supplied to the system and after any register check.
- An attachment SIO to disable the attachment causes a hardware reset of the attachment.

ATTACHMENT LOAD I/O (LIO)

(For use with display adapter only)

Op Code (hex)	Q-Byte (binary)	Operand Address	
Byte 1	Byte 2	Byte 3	Byte 4
31	xxx x xxx	Operand 1 address	
71	xxx x xxx	Op 1 disp from XR1	
B1	xxx x xxx	Op 1 disp from XR2	

DA M N

Q-Byte in Hex	Destination
48 through 4F	32 high-density buffers (Figure 10-13)
58	Control storage
59	Op decode registers
5A through 5F	Invalid

Hex 31, 71, and B1 specify a load I/O operation.

Operation

This instruction transfers 2 bytes of data from the main storage field specified by the operand address to the processing unit local storage registers or to the destination in the display adapter attachment, as specified by the Q-byte.

Program Notes

- An LIO specifying the attachment local control storage must be issued to each position in control storage after power up in order to load the microprogram.
- An LIO specifying the attachment op decode register must be issued to each op decode register after power up to initialize the attachment.

ATTACHMENT TEST I/O AND BRANCH (TIO)

(For use with display adapter only)

Op Code (hex)	Q-Byte (binary)	Operand Address	
Byte 1	Byte 2	Byte 3	Byte 4
C1	0101 1 xxx	Operand 1 address	
D1	0101 1 xxx	Op 1 disp from XR1	
E1	0101 1 xxx	Op 1 disp from XR2	

DA M N

N-Code	Condition Tested
000	Attachment not ready
010	High density buffer/external check (diagnostic); also sets attachment check
011	Control storage check (diagnostic); also sets attachment check
100	Storage address check (diagnostic); also sets attachment check
101	Attachment check (diagnostic)
110	Storage write check (diagnostic); also sets attachment check

Any N-code not shown is invalid and causes:
- Program check if interrupt level 7 is enabled on Model 15
- Processor check if interrupt level 7 is not enabled on Model 15
- Processor check on Models 8 and 12

M-code must be 1.

Hex 5 specifies the display adapter attachment as the addressed unit.

C1, D1, or E1 specifies a test I/O and branch operation. The first hex character in the op code specifies the type of operand addressing for the instruction.

Operation

This instruction tests the display adapter attachment for the condition or conditions specified by the N-code. A condition met response causes the program to branch to the main storage location specified by the operand address. If the condition does not exist, the program accesses the next sequential address and continues.

ATTACHMENT ADVANCE PROGRAM LEVEL (APL)

(For use with display adapter only)

Op Code (hex)	Q-Byte (binary)	R-Byte (binary)
Byte 1	Byte 2	Byte 3
F1	0101 1 xxx	0000 0000

DA M N — This byte is not used for an APL instruction.

N-Code	Condition Tested
000	Attachment not-ready
010	High density buffer/external check (diagnostic); also sets attachment check
011	Control storage check (diagnostic); also sets attachment check
100	Storage address check (diagnostic); also sets attachment check
101	Attachment check (diagnostic)
111	Storage write check (diagnostic); also sets attachment check

Any N-code not shown is invalid and causes:
- Program check if interrupt level 7 is enabled on Model 15
- Processor check if interrupt level 7 is not enabled on Model 15
- Processor check on Models 8 and 12

M-code must be 1.

Hex 5 specifies the display adapter attachment as the addressed unit.

F1 specifies an advance program level operation. F as the first hex character in the op code specifies a command-type instruction (that is, an instruction without operand addressing).

Operation

This instruction tests for the conditions specified in the Q-byte.

- Condition present:
  - Systems with Dual Program Feature installed and enabled, activate the inactive program level.
  - Systems without Dual Program Feature installed or with Dual Program Feature installed but not enabled, loop on the advance program level instruction until the condition no longer exists.

- Condition not present: Systems with or without Dual Program Feature access the next sequential instruction in the active program level.

Program Note

For additional information concerning the Advance Program Level instruction, see Chapter 2.

ATTACHMENT SENSE I/O (SNS)

(For use with display adapter only)

Op Code (hex)	Q-Byte (binary)	Operand Address	
Byte 1	Byte 2	Byte 3	Byte 4
30	xxx x xxx	Operand 1 address	
70	xxx x xxx	Op 1 disp from XR1	
B0	xxx x xxx	Op 1 disp from XR2	

DA M N

Q-Byte in Hex	Source
48 through 4F	32 High-density buffers (see Figure 10-13)
58	Control storage
59	Op decode registers
5A through 5F	Invalid (cause processor check or program check)

Hex 30, 70, and B0 specify sense I/O operations. The first hex character in the op code specifies the type of operand addressing for the instruction.

Operation

Sense attachment I/O transfers 2 data bytes from the source specified by the Q-byte to the main storage field specified by the operand address.

Program Notes

- The attachment is never busy to a sense I/O command.
- Attachment SNS commands must not be issued until the op decode registers have been loaded.
- To ensure that control storage has been loaded correctly, issue an attachment sense command specifying control storage as the source.

Q-Byte

HDB

Q-Byte (hi)	Q-Byte (lo)	HDB bit position 0 1 2 3 4 5 6 7
0	0	Message buffer adr reg (hi)
0	1	Message buffer adr reg (lo)
0	2	Microprogram dependent
0	3	
0	E	
0	F	Cycle steal
1	0	Microprogram dependent
1	E	
1	F	MIAR (lo)
2	0	Microprogram dependent
2	5	
2	6	SIO IR
2	7	SIO IQ
2	8	Microprogram dependent
2	F	
3	0	Microprogram dependent
3	1	Microprogram dependent
3	2	Stop adr reg (hi)
3	3	Stop adr reg (lo)
3	4	Tx adr reg (hi)
3	5	Tx adr reg (lo)
3	6	SIO IR
3	7	SIO IQ
3	8	BSCA status (hi)
3	9	BSCA status (lo)
3	A	Microprogram dependent
3	D	
3	E	Link adr (hi)
3	F	Link adr (lo)

Bit Position

Figure 10-13. High Density Buffer Addressing

SERIAL INPUT/OUTPUT CHANNEL ADAPTER (SIOC)

The System/3 serial input/output channel adapter (SIOC) provides attachment circuitry for such additional input/output devices as the 1255, 1270, 1419 and the 3881. These devices are described in this chapter. The control unit of any I/O unit that is to be attached to the SIOC must be designed to be compatible with the SIOC. Only one control unit can be physically attached to the SIOC at any one time, although more than one I/O device can be controlled by that control unit. If the control unit is controlling more than one device, only one device can operate at any time. The SIOC handles data in the form of an 8-bit byte (plus parity). Data is transferred 1 byte at a time, parallel by bit.

The SIOC provides an intermediate control unit between the system I/O channel and the device control unit. This intermediate control unit produces the necessary signals to control the device control unit from information furnished to the SIOC by instructions from the processing unit, control bytes stored in registers in the SIOC by the processing unit, and information supplied by the device control unit.

SIOC Data Transfer Register

A 9-bit data transfer register is provided in the SIOC to temporarily store 1 byte of data (8 bits plus parity) that is to be transferred between the I/O device and main storage. Data transfer is normally on a cycle steal basis, but the contents of this register can be moved between the register and main storage with load I/O and sense I/O instructions when this is required by the characteristics of the I/O device involved or for diagnostic purposes. The register is tested for correct parity; a sense bit is set by incorrect parity.

SIOC Data Address Register

This is one of the local storage registers used to store the address of the data field that is to be used by the I/O device. The register is loaded by a load I/O instruction and can be sensed by a sense instruction.

SIOC Length Count Register

Because data transfer occurs on a cycle steal basis, the SIOC must keep track of the number of bytes transferred. A length count register is provided to perform this function. This counter limits the number of bytes to be transferred to 256 bytes per record. A load I/O instruction is used to place the number of bytes to be transferred in the length count register. The number that is placed in the length count register is the binary representation of a number equal to 256 minus the number of bytes to be transferred. Normally, the I/O device signals when enough bytes have been transferred, but the length count register signals when the correct number of bytes has been transferred, and prevents further data transfer. The contents of this register and the count-exceeded condition can be placed in storage with a sense I/O instruction.

I/O Select Register

This register is used for issuing up to 16 separate I/O device control signals. It is loaded with a start I/O instruction. The functions that these control signals perform in the I/O device are determined by that device, and the data that must be placed in the register for differing conditions is defined by that device. In general, they will not all be used by any one device.

I/O Transfer Lines

This is not a hardware register but a set of 11 signal lines from the device to the SIOC that communicates information to the processing unit. These lines can be tested with the sense I/O instruction and used for program decisions based on information received from the I/O unit. The conditions that are conveyed by these lines are defined by the I/O devices and are specified in manuals, or sections of this manual, relating to the I/O device.

The transfer lines are bit significant as follows:

Low-Order Byte (N-Code 011)

Bit	Meaning
0	I/O transfer line 8
1	I/O transfer line 7
2	I/O transfer line 6
3	I/O transfer line 5
4	I/O transfer line 4
5	I/O transfer line 3
6	I/O transfer line 2
7	I/O transfer line 1

High-Order Byte (N-Code 011)

Bit	Meaning
0	I/O identifier bit 8
1	I/O identifier bit 4
2	I/O identifier bit 2
3	I/O identifier bit 1
4	I/O device attached
5	I/O transfer line 11
6	I/O transfer line 10
7	I/O transfer line 9

Function Register

This register defines the mode of operation of the I/O device. It must be loaded before attempting to execute the program operating the device (it can be loaded by that program before any device operations are attempted). The specific bits that must be stored in this register by a load I/O instruction are defined by the I/O device.

The bytes loaded into the function register are bit significant as follows:

Low-Order Byte (operand address)

Bit	Meaning
0	Diagnostic mode (used only for CE diagnostic testing)
1	Spare
2	Latch transfer line 4
3	Latch transfer line 3
4	Latch transfer line 1
5	Transfer line 3; reset disconnect latch
6	Reset disconnect latch after 6 microseconds
7	Transfer line 5; reset disconnect latch

High-Order Byte (operand address minus 1)

Bit	Meaning
0	Write mode set service response
1	Reset service response after 6 microseconds
2	Transfer line 2 EOT
3	Transfer line 1 EOT
4	Even parity
5	Decrement DAR
6	Latch I/O 1 select
7	Slave (transfer lines 6 and 7 latch control)

SIOC Operation

The operation of the SIOC requires that certain I/O instructions be performed to prepare the program and adapter for operation. A means of identifying the individual I/O devices that are attached to the SIOC is provided at the time the I/O device is designed. Four identification lines are stored on a sense I/O operation that specifies the byte that contains their sense bits. The following procedures should be performed to operate the SIOC:

1. Sense the I/O identification byte.

2. Test that an I/O device is attached to the SIOC.

3. Test which I/O device is attached to the SIOC.

4. Load the function register with the appropriate bytes to control the particular I/O device.

I/O operations require that certain instructions be performed before executing the instruction that transfers data. Before each data transfer operation, the length count register must be loaded with a count equal to 256 minus the number of bytes to be transferred by that operation. The SIOC data address register must be loaded with the address of the first byte of the data field to be operated on. Then the start I/O instruction that transfers the data can be issued. Testing and sensing operations should be included in the operating program but can be inserted at the discretion of the programmer in accordance with good programming practice.

The SIOC operates in interrupt mode on interrupt level 4. Each time the I/O device requires some special service from the processing unit, such as processing in time for stacker selection, it interrupts the processing unit. Interrupts must be enabled for the I/O device before the SIOC can interrupt the processing unit.

The commands for all I/O devices attached to the SIOC are the same; the interpretation given to some of the commands by the I/O devices may be different. The interpretations are discussed in the I/O device sections.

Status and Diagnostic Bytes

These bytes are included in the sense I/O instruction and are the high-order bytes of their respective sense operations.

Bits 0 and 1 of the status byte and all of the bits of the diagnostic byte are for CE diagnostic use and have no meaning to the I/O control program.

The status and diagnostic bytes are bit significant as follows:

Status Byte (N-Code 010)

Bit	Meaning
0	Spare
1	End request
2	Interrup pending
3	I/O attention
4	Data transfer register parity check
5	No-op
6	Length count register overflow
7	I/O ready

Diagnostic Byte (N-Code 101)

Bit	Meaning
0	SIOC interrupt request latch
1	Service request
2	Service response
3	Interrupt enable
4	I/O disconnect
5	Write call
6	Read call
7	I/O selected

IBM 1255 MAGNETIC CHARACTER READER

The IBM 1255 Magnetic Character Reader provides the capability of entering data inscribed with magnetic ink characters on paper documents. The 1255 is available in these models:

Model	Font Read	Maximum Throughput[2]	Number of Stackers
1	E-13B	500/min.	6
2	E-13B	750/min.	6
3	E-13B	750/min.	12
21	CMC 7[1]	500/min.	6
22	CMC 7[1]	750/min.	6
23	CMC 7[1]	750/min.	12

(Models 21, 22, and 23 not offered in U.S.A.)

A discussion of the capabilities, characteristics, and operations of the magnetic character reader can be found in *IBM 1255 Magnetic Character Reader Components Description,* GA24-3542.

1255 Special Features

Account Number Checking

For a description of the manner in which account number checking is performed, see the *IBM 1255 Magnetic Character Reader Components Description,* GA24-3542. If an account number is found incorrect when this feature is installed, the account number field valid indicator bit is turned off. No special programming is involved with the account number checking feature.

51-Column Sort Feature

This feature allows the 1255 to handle documents shorter than the standard documents. These documents lack a transit-routing field. This fact could be used by a program to distinguish 51-column documents from others.

[1] Character font used outside the United States
[2] Measured with 6-inch documents

Dash Symbol Transmission

This feature allows the 1255 to transmit the dash symbol from the transit-routing field. Because different nations of the world use the dash symbol in different positions of their transit-routing fields, this fact can be used by programming to distinguish between checks from different countries.

Document Counter

This feature has no effect on programming the 1255 for System/3.

IBM 1270 OPTICAL READER SORTER

(IBM 1270 is not offered in U.S.A.)

The IBM 1270 Optical Reader Sorter reads OCR characters from paper documents and transmits them to the processing unit. A discussion of the capabilities, characteristics, and operations of the 1270 optical reader sorter can be found in *IBM System/360 Component Description—IBM 1270 Optical Reader Sorter,* GA19-0035.

Feeding Documents on 1270

The engage command starts the flow of documents if the 1270 is online and is in a ready-to-feed state.

A disengage command stops document feeding by stopping the separator. Document feeding also stops whenever the processing unit is stopped. The following 1255 conditions also stop or inhibit document feeding:

1. The separator stops feeding documents but the transport continues to run whenever:
   a. The start key is being held down.
   b. The stacker is full.
   c. No validity-check-and-readout key has been pressed.

2. The transport mechanism stops because of:
   a. Stop key depression
   b. An empty hopper
   c. A feed jam
   d. A transport jam
   e. A sort check
   f. An interlock

When the 1270 is online and not-read-to-feed, the engage command is stored in the 1270 so that the actual document feeding starts as soon as the 1270 becomes ready-to-feed again.

If a disengage instruction is issued when the 1270 is online and not ready-to-feed, no documents are fed when the 1270 is returned to the ready-to-feed state.

The system call light on the 1270 operator panel indicates that the program calls for document feeding.

The engage command is issued by executing a start I/O instruction for the SIOC with an N-code of 100, and a control code of 00000001. The disengage command is issued by a start I/O instruction with an N-code of 100 and a control code of 00000010.

Note: An engage instruction immediately followed by a disengage instruction causes single-document feeding.

Retrieving Data from 1270 Documents

A start I/O instruction specifying read retrieves data from documents passing through the 1270. A read command must be issued for each document before that document reaches the read head. Failure to issue the read command in time results in the document being rejected by the 1270 and a signal (auto-select) being provided for program interrogation.

Validity check and readout keys on the 1270 operator panel select the data to be transferred to system main storage. The first character transferred from the 1270 enters storage at the address designated by the SIOC data address register. Subsequent characters enter successively lower storage locations.

The read operation is terminated either at the end of each document or when the specified number of characters to be read (as initially loaded into the SIOC length count register), have been transferred, whichever occurs first. At the end of a read operation the SIOC requests an interrupt to process the data and select the appropriate stacker (pocket) for that document.

Directing the Disposition of 1270 Documents

Stacker select commands direct documents to the stackers in the 1270. These commands are generated by start I/O instructions that load the I/O select register.

The stacker select command must be issued within 24 milliseconds after the document to be selected leaves the read head. If a stacker select command has not been issued within this time, a sort check occurs, the 1270 rejects the document, and the 1270 stops after all documents in the transport have been directed to the reject stacker.

The sort check light on the 1270 operator panel indicates the error to the operator and signals (sorter-is-stopped and auto-select) are provided for program interrogation.

Termination of 1270 Operations

The following rules must be observed in order to prevent nonrecoverable 1270 and/or system errors:

- Do not stop the processing unit during 1270 online operations.
- Do not switch from online mode to offline mode or vice versa during 1270 operations.
- The 1270, when attached to the system, should be powered down only when there is no activity on the serial I/O channel interface.

While system-is-stopped light (on the 1270 operator panel) is on, there is no SIOC interface activity.

IBM 1419 MAGNETIC CHARACTER READER

The IBM 1419 Magnetic Character Reader can be attached to System/3 via the SIOC. Programming for the 1419 is basically the same as programming for a 1255 attached to the system. However, the following considerations and features apply for the 1419.

- The I/O ATTENTION light indicates that the 1419 has stopped because the stop delay has timed out, because of a left feed jam, or because of a right feed jam.
- The 1419 uses the same ID code that is used for the 1255.
- The 1419 provides the processing unit with 8-bit EBCDIC code with odd parity.
- The 1419 provides a minimum of 9.5 milliseconds between documents. The program must determine pocket selection and update batch numbering in this time.
- The 1419 is a 13-pocket (A, B, 0-9 and REJECT) device. The 1255 is a 6- or 12-pocket device and does not have the equivalent of a B-pocket. Pocket selection is the same on a 1419 as for the 1255 for pockets A, 0-9, and REJECT. Pocket selection for the B-pocket is via SIO instruction I/O control byte 2 (N-code = 100), bit 2.
- To satisfy the batch numbering and programmable pocket light features available for the 1419, the 1419 program must control a 1419 KN-latch-on function. To turn the KN latch on and off, use an SIO control instruction (N-code 100) and set control byte 2, bit 0 on or off.
- Batch numbering update is an online feature only. To update the batch numbering feature, first issue a KN latch on instruction, followed by a pocket reject instruction. The two instructions must be issued within the same 9.5 ms interval and in the sequence stated. The sequence is in addition to and independent of pocket selection.
- Programmable pocket lights is an online feature only. It is available in two groups of six pockets per group. The first group is associated with pockets A, B, 0-3; the second group is associated with pockets 4-9. The first group is a prerequisite of the second group. The following sequence is required for proper operation:

  1. Count the documents going to each pocket; and when any pocket reaches a predetermined number, issue a disengage instruction. (Consider documents in process.)
  2. Issue the appropriate pocket select instruction for each document in process after issuing the disengage instruction.
  3. After all documents in process have been routed correctly, issue a KN-latch-on instruction.
  4. Issue pocket select instructions for the pocket or pockets that contain the specified number of documents.

     Note: Because the operator turns the KN latch off by pressing the foot treadle, it is advisable to issue a KN-latch-on instruction before issuing each pocket select instruction. This ensures against accidental KN-latch reset when more than one pocket light will be turned on after any one disengage function.
  5. Reset the document counters in the processing unit for those pockets that were selected in step 4.
  6. The program can now enter a loop by issuing an engage instruction and waiting for the arrival of a document to be read. No documents will feed while the KN latch is on.
  7. Press the foot treadle and remove the documents from the lighted pocket or pockets. When the foot treadle is released, the KN latch resets and document feeding resumes.

- Dash symbol transmission from the transit-routing field is an online feature only. Each 1419 equipped with this feature transmits the dash in the transit-routing field to the processing unit.
- The split field feature is basically an offline feature that allows the splitting of variable-length fields into two elements by use of a dash. However, if the split field feature is installed, dashes from variable-length fields are transmitted to the processing unit during online mode operations.
- Printout on a 5203 depends on chain and train used. The standard character set does not include a special symbol 3 (SS3).

SIOC PROGRAMMING REQUIREMENTS FOR 1255, 1270, AND 1419

In addition to the instructions that control functions of the reader, the following items must be handled in a specific manner in order for the reader to operate with the SIOC:

1. Before executing the instructions that cause the reader to operate, the function register of the SIOC must be loaded by a load I/O instruction. The 2 bytes loaded must contain a 1 in bits 1 and 5 of the high-order byte and a 1 in bit 6 of the low-order byte. All other bits in these bytes must be 0.

2. The length count register must be loaded by a load I/O instruction issued to the SIOC. The number to be loaded into the register is 256 minus the number of bytes to be read. This operation must be performed before each read instruction.

3. The SIOC data address register must be loaded, (by a load I/O instruction) with the address of the low-order byte of the data field before each read operation. The address designates where to store the data read from the document.

4. The device identification assigned to the SIOC is 0011. The fact that the reader is the device attached to the SIOC is detected by the sense I/O instruction sensing the I/O transfer lines. Bits 0 through 3 of the high-order sense byte stored by this instruction contain the device identification. For the reader, bits 0 and 1 are 0 and bits 2 and 3 are 1.

5. A start I/O instruction must be issued to enable interrupts for the SIOC. The reader requires that processing for the documents be performed within specified periods of time to provide correct processing. The reader causes an interrupt at the end of every document, and this interrupt must be enabled to allow processing to start.

IBM 3881 OPTICAL MARK READER

An IBM 3881 Model 1 Optical Mark Reader can be attached to the serial input/output channel to provide direct data entry from mark read data forms to System/3. The *IBM 3881 Optical Mark Reader Models 1 and 2 Reference Manual and Operator's Guide,* GA21-9143, describes the 3881, its internal microprocessor, and some of its applications. It also explains the theory and use of mark read data input forms, describing how to adapt the microprocessor's built-in error detection logic and mark translation ability to the exact requirements of each data field being read. It gives complete details about keys, lights, error indications, and operating procedures, and contains forms specifications and marking recommendations.

3881 Output Record

The 3881 assembles an output record that contains a segment descriptor word, record descriptor word, normal mark data bytes, BCD data bytes (if any), and serial number data (if any) in the format shown in Figure 11-1 and 11-2. The output record moves to the system under system control.

Segment Descriptor Word

The first 2 bytes indicate (in binary code) the number of bytes in the record. The last 2 bytes contain 0's.

Record Descriptor Word

Contains 2 bytes of status information that indicate whether the form was selected by the 3881, whether the record contains a serial number, etc (Figure 11-2).

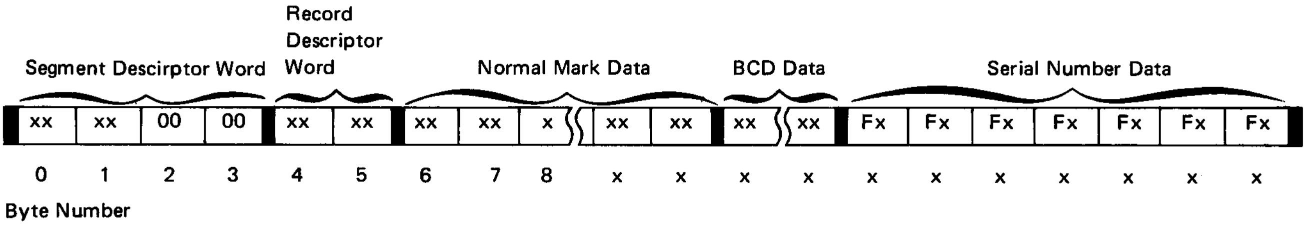


Figure 11-1. Format of 3881 Output Record

First Byte of Descriptor Word (Byte 4 of Output Record):

Value in Byte 4		Conditions Indicated by Hexadecimal Value			
Hex	EBCDIC	Serial Numbering Feature Used	3881 Selected Form to Select Stacker	Reject Characters on Form	BCD Data Length Error
C8	H	No	No	No	No
C1	A	No	No	No	Yes
C2	B	No	No	Yes	No
C3	C	No	No	Yes	Yes
C4[1]	D	No	Yes	No	No
C5	E	No	Yes	No	Yes
C6	F	No	Yes	Yes	No
C7	G	No	Yes	Yes	Yes
F0	0	Yes	No	No	No
F1	1	Yes	No	No	Yes
F2	2	Yes	No	Yes	No
F3	3	Yes	No	Yes	Yes
F4[1]	4	Yes	Yes	No	No
F5	5	Yes	Yes	No	Yes
F6	6	Yes	Yes	Yes	No
F7	7	Yes	Yes	Yes	Yes

[1]Unused

Second Byte of Descriptor Word (Byte 5 of Output Record):

Value in Byte 5		
Hex	EBCDIC	Format Type of Form
F0	0	Basic
F1	1	Alternate 1
F2	2	Alternate 2
F3	3	Alternate 3
F4	4	Alternate 4
F5	5	Alternate 5

Figure 11-2. 3881 Record Descriptor Word

Normal Mark Data Field

Contains all the bytes of data from regular mark data fields on the form. The type of data in each byte (digit, letter, or image format bit), the sequence in which each field is loaded into the normal data field of the record, and the number of data bytes in each field is controlled by a microprocessor in the 3881.

BCD Data Field

This field is in the output record only if the 3881 BCD special feature is installed and the 3881 has been controlled (via a format control sheet) to read BCD data from that form. The field contains 1 EBCDIC byte for each BCD-encoded digit read. BCD fields are loaded into the output record BCD area (at the end of the normal mark data field, as shown in Figure 11-1) in the sequence they were read from the form.

Serial Number Data Field

This field contains a 7-digit decimal number for batch and/or serial numbering identical to the number printed on the input form while it passed through the 3881 transport. If the serial numbering device (special feature) is not installed on the 3881, or if it is installed and is not active when the form is read, the 3881 does not place a serial number on the output record. If the device is installed and active for one type of form only (those directed by the 3881 to the select stacker, for example), the 3881 loads blanks into the serial number field for all unnumbered forms. The decision to number or not to number a particular form is a function of 3881 logic only, and is not affected by processing unit programming control over form selection.

3881 Operations

Operator action during initial 3881 setup procedures causes a forms feed cycle. The first data form moves past the read heads, is read by the 3881, and stops at the wait station awaiting a stacker selection decision. As the form passes the read head, the 3881 microprocessor converts the marks into meaningful data bytes and formats them into an output record in the 3881 read buffer. At the same time, the 3881 error detection and mark discriminating logic (using parameters stored in the 3881 microprocessor for the type of form being read) selects the select stacker for the form if it contains an error or questionable mark.

The microprogram builds record descriptor bytes for each form it reads. These are bytes 4 and 5 of the output record, and they indicate whether the 3881 logic directed the form to the select stacker, serial numbering was enabled, invalid marks were detected, etc (Figure 11-2). Record descriptor bytes never reflect programmed stacker selection.

After the first form is fed and read by the 3881, the 3881 enters a ready status. The program must now issue one or more read commands to move the data from the 3881 buffer to the processing unit. (Each System/3 read command moves a maximum of 256 bytes from the 3881 buffer.)

An SIO specifying a read operation places the SIOC in the read mode; this permits the transfer of data serially by byte from the 3881 read buffer to the processing unit. Data transfer starts immediately if the entire output record was formatted in the 3881 read buffer (signalled by the 3881 setting the output record ready bit).

If the command is issued after a feed command was issued but before the output record is completely loaded into the read buffer, the 3881 accepts the command, and starts data transfer when the output record ready bit comes on.

If the command is issued after data transfer occurred for one record and before a feed command is issued to load the buffer with data from the next form, the SIOC returns a busy status indication, and the 3881 ignores the command. To recover, issue an SIO with an N-code of 000 or 001 and a control code with bit 4 on; this turns the busy bit off.

The processing unit stores the first byte of data from the 3881 in the storage location specified by the SIOC data address register, storing subsequent bytes in successively higher storage locations. Data transfer continues until the read operation is stopped by (1) a length count register overflow or (2) an end of data transfer (EOT) condition, which indicates that the last byte was moved from the 3881 buffer to the system.

If more than 256 bytes are to be read from the buffer, the program must issue multiple read commands to move the entire output record from the buffer to the processing unit.

If the programmer knows the exact number of bytes to transfer for each record, he can specify a certain number (up to 256) of bytes to move per read command, and issue the number of read commands necessary to move all the data to the processing unit. If the sum of the bytes specified by all the read commands for that output record equals the exact number of bytes in the output record, all the data is transferred to the system.

Often, however, the programmer will not know the exact number of bytes in the output record to be transmitted to the system. In this case, he can load hex 00 into the length count register, then test for an output record ready status bit at the end of the read operation. If the bit is on, there is still data to transfer to the processing unit, so he must issue another read command with hex 00 in the length count register. All the data is transmitted to the system when the program detects the output record ready status bit off.

The program may need to retransmit data from the 3881 read buffer to the processing unit before that data is destroyed by reading new data into the buffer with a feed instruction. An SIO command specifying the reread enable function can be issued any time after the output record ready bit is turned on and before the next feed command is issued. The reread enable command allows the program to retransmit data from the buffer, starting with the first byte of the output record and performing the identical read functions described earlier.

The feed and select stacker command must now be issued to move a new form past the 3881 read station and place a new output record in the 3881 buffer. This feed command forces the form whose data was last transferred to the processing unit to move to a stacker. Unless the 3881 specified the select stacker when the program specified the normal stacker for the same form, the form enters the stacker specified by the processing unit feed and select stacker command. Each form stops at the wait station awaiting a stacker select decision if the feed and select stacker command is not issued before the form reaches the wait station. Any subsequent feed and select stacker command issued without an intervening read command is ignored by the 3881. This ensures that no data from a form can be destroyed by data from the next form until it is transferred to the system.

Code Representing Invalid Combination of Marks on 3881 Forms

Whenever the 3881 recognizes an invalid combination of marks on a form, it inserts a special code (instead of the invalid data) in the output record. For this code, the customer can specify either the standard hex 3F or (as a no-cost option) hex 7C, which is a printable graphic. Hex 3F is not a printable EBCDIC character, whereas hex 7C causes the @ character to print from most IBM print chains.

I/O Attention Light on the Processing Unit

The 3881 does not use the I/O attention light on the processing unit.

1255/1270/3881 START I/O (SIO)

Op Code (hex)	Q-Byte (binary)	R-Byte (binary)
Byte 1	Byte 2	Byte 3
F3	0011 0 xxx	xxxx xxxx

DA M N — Control Code Bits

N-Code	Bits 0123 4567	Function Specified		
00x	0000 0001	Reset interrupt request (performed by SIOC)		
	0000 0010	Enable interrupt request (performed by SIOC)		
	0000 0100	Disable interrupt request (performed by SIOC)		
	0000 1000	Reset SIOC adapter, removing SIOC from busy state (performed by SIOC)		
	0001 0000	Set interrupt request		
001	0000 0000	Read		
		1255 Models 1, 2, 21, 22	*1255 Models 3, 23 and 1270*	*3881*
011	1000 0000	Not used	Select stacker 7	Not used
	0100 0000	Select stacker 6[1]	Select stacker 6	Not used
	0010 0000	Not used	Select stacker 5	Not used
	0001 0000	Select stacker 4	Select stacker 4	Not used
	0000 1000	Select stacker 3[2]	Select stacker 3	Not used
	0000 0100	Select stacker 2	Select stacker 2	Enable reread
	0000 0010	Select stacker 1[2]	Select stacker 1	Feed and select select stacker
	0000 0001	Select stacker 0	Select stacker 0	Feed and select normal stacker[3]
100	1000 0000	Not used	Not used	Not used
	0100 0000	Select reject stacker	Select reject stacker	Not used
	0010 0000	Not used	Not used	Not used
	0001 0000	Not used	Select stacker A	Not used
	0000 1000	Not used	Select stacker 9	Not used
	0000 0100	Select stacker 8[1]	Select stacker 8	Not used
	0000 0010	Disengage feed	Disengage feed	Not used
	0000 0001	Engage feed	Engage feed	Not used

Any N-code not shown is invalid and causes:
- Program check if interrupt level 7 is enabled on Model 15
- Processor check if interrupt level 7 is not enabled on Model 15
- Processor check on Models 8, 10, and 12

The M-code is always 0 for a device attached to the SIOC.

The device address is always hex 3 for a device attached to the SIOC.

F3 specifies a start I/O operation. F, as the first hex character in the op code, specifies a command-type instruction (that is, an instruction with no operand addressing).

---

[1] Invalid code for optional (0-4/5-9) sort pattern readers

[2] Invalid code for standard (even/odd) sort pattern readers

[3] The 3881 microprocessor can override this selection, sending the form to the select stacker because it detected a 3881 format-control-sheet-controlled forms selection condition. The stored program can examine byte 4 of the output record to determine whether or not the 3881 sent the form to the select stacker.

Operation

The start I/O instruction is used to control the mode of operation of the SIOC adapter, and to issue control signals (I/O select lines) to the attached I/O device. A start I/O read or write instruction electronically attaches the adapter to the I/O device by setting the device in either the read or write mode. The attachment must be placed in either one of these modes for the transfer of data to occur. This instruction is also used to enable or disable the ability of the adapter to request an interrupt priority, if required by the attached I/O device. If an interrupt is requested and the interrupt ability is disabled, the interrupt is kept pending in the SIOC adapter. The interrupt pending condition can be program interrogated by a sense instruction. The interrupt request is reset and the SIOC adapter is also removed from the busy state with the SIO instruction.

SIO instructions with N-code 000 are accepted and executed by the adapter, regardless of its operating status; SIO instructions with N-codes 011 or 100 are accepted and executed by the adapter unless a busy condition exists. A busy condition causes the instruction to be rejected and causes the program to loop at the SIO instruction until it can be accepted. When the adapter becomes not busy the instruction is accepted and normal instruction sequencing continues.

If the processor is not executing an SIOC interrupt routine, SIO instructions with N-codes 001 or 010 are accepted and executed by the adapter unless an I/O attention or busy condition exists. In these cases the instruction is rejected as described in the preceding paragraph. When the adapter becomes not busy, or when the cause of the I/O attention condition is removed, the instruction is accepted and the normal instruction sequence continues.

If an SIO instruction with N-code of 001 or 010 is issued when a device is not attached, the instruction cannot be executed. The processing unit sets the no-op bit without executing the SIO instruction. This status bit can be sensed and reset with a SNS instruction.

If an SIO instruction with N-code 001, 010, 011, or 100 is issued and the no-op status bit is active, the instruction is accepted but is not executed and the no-op status bit remains active.

If an SIO instruction with N-code 001 or 010 is issued during an SIOC interrupt routine and the I/O attention signal is active, the instruction is accepted but is not executed. The no-op status bit is set and the program advances to the next sequential instruction. This prevents processing unit hangup as a result of the I/O attention signal becoming active during the SIOC interrupt routine. The ability to issue and execute SIO instructions with an N-code of 000 permits programming to recover from this situation. A reset interrupt request instruction can be used to exit the interrupt routine.

Combinations of the N-code not shown in this section are invalid. Figure 11-3 summarizes SIOC operations according to the adapter status.

Program Note

- Stackers on the 12-stacker readers are arranged in two vertical rows of six stackers each. Stackers on the left blank are numbered, from bottom to top: 0, 1, 2, 3, 4, and R. Those on the right bank are numbered 5, 6, 7, 8, 9, and A.
- Not all the stackers indicated in the preceding note will be available on a six-pocket (six-stacker) 1270. A six-pocket 1255 or 1270 will be ordered with either stacker designations 0 through 4 and R (for reject stacker) or with the even-numbered stacker designations 0, 2, 4, 6, 8, and R.

Instruction[1]	DBO Parity Error	Device Not Attached	Busy	I/O Attention Interrupt Routine	I/O Attention Not Interrupt Routine	No-Op Bit On	DTR Parity Check[2]
SIO							
N-code 0	Processor check	Execute	Execute	Execute	Execute	Execute	Execute
N-code 1	Processor check	No-op	Reject	No-op	Reject	No-op	Execute
N-code 2	Processor check	No-op	Reject	No-op	Reject	No-op	Execute
N-code 3	Processor check	Execute	Reject	Execute	Execute	No-op	Execute
N-code 4	Processor check	Execute	Reject	Execute	Execute	No-op	Execute
LIO (all valid N-codes)	Processor check	Execute	Reject	Execute		No-op	Execute
SNS (all valid N-codes)	Processor check	Execute	Execute	Execute		Execute	Execute
TIO							
N-code 0	Processor check	Branch	Not applicable	Branch		Branch	Branch
N-code 2	Processor check	Not applicable	Branch	Not applicable		Not applicable	Not applicable

[1]An invalid instruction causes a processor check, stopping the system.
[2]The data transfer register parity check status bit is reset when the adapter recognizes a valid SIO instruction.

Note: When the adapter performs a no-op routine, it accepts the instruction but does not execute it.

Figure 11-3. Summary of Instruction Handling Based Upon Adapter Status

Checking

The contents of the data transfer register and the I/O channel data are tested for parity errors during data transfer operations and whenever instructions or data is being transmitted over the I/O channel to the SIOC adapter. Detected parity errors on data coming from the processing unit result in a processor check stop with a parity error indication. Parity errors detected in the data transfer register set a data transfer register parity check sense bit that can be tested by a sense I/O instruction.

Op Code (hex)	Q-Byte (binary)	Operand Address	
Byte 1	Byte 2	Byte 3	Byte 4
C1	0011 0 xxx	Operand 1 address	
D1	0011 0 xxx	Op 1 disp from XR1	
E1	0011 0 xxx	Op 1 disp from XR2	

DA M N

N-Code Condition Tested

N-Code	Condition Tested
000	Not ready/check
010	Busy

Any N-code not shown is invalid and causes:
- Program check if interrupt level 7 is enabled on Model 15
- Processor check if interrupt level 7 is not enabled on Model 15
- Processor check on Models 8, 10, and 12

M-code is always 0 for devices attached to the SIOC.

The DA-code of any device attached to the SIOC is always hex 3.

C1, D1, or E1 specifies a test I/O and branch operation. The first hex in the op code specifies the type of operand addressing to be used for the instruction.

Operation

The processing unit tests the conditions specified by the N-code. If the condition is present, the processing unit places the address from the instruction address register into the address recall register, then places the operand address from the instruction into the instruction address register, and then accesses that instruction. If the condition tested is not present, the processing unit places the operand address from the instruction in the address recall register and executes the address specified by the instruction address register (the next sequential address).

Program Notes

- The not-ready/check TIO condition indicates that one or more of these conditions exist:
  - No I/O device is attached to the SIOC.
  - The data transfer register has incorrect parity.
  - A read or write SIO instruction addressed to the SIOC resulted in a no-op condition.
  - An I/O attention condition exists at the attached device.

- The busy TIO condition indicates that the I/O device attached to the SIOC is performing an operation.

1255/1270/3881 ADVANCE PROGRAM LEVEL (APL)

Op Code (hex) Byte 1	Q-Byte (binary) Byte 2	R-Byte (binary) Byte 3
F1	0011 0 0xx	0000 0000

DA M N

R-byte is not used in an APL instruction.

N-Code	Condition Tested
000	SIOC not-ready/check
010	SIOC busy

Any N-code not shown is invalid and causes:

- Program check if interrupt level 7 is enabled on Model 15
- Processor check if interrupt level 7 is not enabled on Model 15
- Processor check on Models 8, 10, and 12

The M-code for any device attached to the SIOC is always 0.

The device address of any device attached to the SIOC is always hex 3.

F1 specifies an APL operation. F, as the first hex character in the op code, identifies a command-type instruction (that is, an instruction without operand addressing).

Operation

The processing unit tests the device attached to the SIOC for the condition specified by the N-code. If any tested condition exists, the program loops on the APL instruction until the condition no longer exists, then advances to the next sequential instruction. If no tested condition exists, the processing unit immediately accesses the next sequential instruction.

Program Note

If the DA and M-portions of the Q-byte are binary 00000, the APL instruction is treated as a no-op command and the processing unit immediately accesses the next sequential instruction. For this unconditional no-op, the N-code should be binary 000.

1255/1270/3881 LOAD I/O (LIO)

Op Code (hex)	Q-Byte (binary)	Operand Address	
Byte 1	Byte 2	Byte 3	Byte 4
31	0011 0 xxx	Operand 1 address	
71	0011 0 xxx	Op 1 disp from XR1	
B1	0011 0 xxx	Op 1 disp from XR2	

DA M N

N-Code **To Be Loaded**

001 I/O function register

The following two bytes must be loaded into the register:

Device Attached to SIOC	*Hex Characters that Must be Loaded Leftmost Byte*	*Rightmost Byte*
1255	44	02
1270	04	02
3881	40	02

010 SIOC length count register

100 SIOC data address register

101 Data transfer register

Any N-code not shown is invalid and causes:

Program check if interrupt level 7 is enabled on Model 15

Processor check if interrupt level 7 is not enabled on Model 15

Processor check on Models 8, 10, and 12

The M-code is always 0 for a device attached to the SIOC.

The device address of a device attached to the SIOC is always hex 3.

31, 71, or B1 specifies a load I/O operation. The first hex character in the op code signifies the type of operand addressing to be used for the instruction.

Operation

The processing unit loads the 2 bytes of data contained in the operand into the register specified by the N-code. The operand is addressed by its low-order (higher numbered) storage position.

1255/1270/3881 SENSE I/O (SNS)

Op Code (hex)	Q-Byte (binary)	Operand Address	
Byte 1	Byte 2	Byte 3	Byte 4
30	0011 0 xxx	Operand 1 address	
70	0011 0 xxx	Op 1 disp from XR1	
B0	0011 0 xxx	Op 1 disp from XR2	

DA M N

N-Code	Data Source
001	I/O function register
010	Length count register[1] and status byte
011	I/O transfer lines and I/O identification. These lines are bit significant as follows:

Low-Order Byte (operand address)

Bit	Meaning *1255 Models 1, 2, 3*	*1270, and 1255 Models 21, 22, 23*	*3881*
0	Auto reject	Auto reject (1255), Auto select (1270)	Not used
1	Serial number field valid	Field 5 valid	3881 ready
2	Transit/routing field valid	Field 4 valid	Equipment check
3	Account number field valid	Field 3 valid	Output record ready
4	Process control field valid	Field 2 valid	Diagnostic not ready
5	Amount field valid	Field 1 valid	Not used
6	Document under read head	Document under read head	Not used
7	Document to be read	Document to be read	End-of-file

High-Order Byte (operand address minus 1)

Bit	Meaning *Models 1, 2, 3*	*Models 21, 22, 23 & 127*	*3881*
0-3	Must be 0011	Must be 0011	Must be 0010
4	1255 attached	Not used	Device is attached
5	Not used	Field 7 valid	Not used
6	Not used	Field 6 valid	Not used
7	Sorter is stopped	Sorter is stopped	Not used

N-Code	Data Source
100	Data address register
101	Data transfer register[1] and diagnostic byte

Any N-code not shown is invalid and causes:
- Program check if interrupt level 7 is enabled on Model 15
- Processor check if interrupt level 7 is not enabled on Model 15
- Processor check on Models 8, 10, and 12

The M-code is always 0 for an SIOC-attached device.

Hex 3 specifies the SIOC-attached device (the 1255/1270) as the device whose registers and lines are to be sensed.

Hex 30, 70, or B0 specifies a sense I/O operation. The first hex digit in the op code indicates the type of operand addressing for the instruction.

---

[1]This byte is stored at the operand address. The associated byte is stored at the operand address minus 1.

Operation

The processing unit moves the data from the lines or registers specified by the N-code into the 2-byte data field specified by the operand address. The operand is addressed by its higher-numbered storage position.

Program Notes for 1255 and 1270

- The operand is always addressed by the low-order (higher storage number) byte.
- This instruction is executed even though the reader is busy or has a not-ready/check condition.
- The diagnostic byte is for CE diagnostics and has no meaning to the I/O control program.
- Figure 11-4 shows the 1255 and 1270 status bits and their meanings.
- Sorter is stopped (bit 7) is turned on when the main motor is stopped. A main motor stop is caused by a jam, a late stacker select, an empty feed hopper, or the stop key being pressed. This bit is turned off by clearing the stop condition and restarting the reader.
- All field valid indicators are conditioned when their respective fields, including bracketing symbols, are read without error, and deconditioned when the leading edge of the next document is sensed at the read head.
- The auto reject indication turns on for any document that is rejected automatically by the reader. This occurs for a read command that is not issued for a document before the document reaches the read head, a short document, an overly long document, or a document spacing error. The indicator turns on when the error condition is detected and stays on until the following document arrives at the read head, except that for a document spacing error the indicator stays on through the second document because both documents are rejected.

  The auto select condition is also set by a feed jam, transport condition, interlock condition or sort check condition. This auto check stays on for all follow-on documents and is reset by the first document arriving at the read head after error recovery and machine restart.

  A stacker-select command need not be issued for an auto reject document; however, if the command is issued, it must specify the select pocket or a sort check will occur.

- The document under read head bit comes on when a document passes under the read head and turns off when the document leaves the read head. It can be used to determine if a document cleared the read head when the read command was terminated before the end of the document. A stacker select command must not be given for the document until the document leaves the read head.
- The document to be read bit is on as soon as the reader tries to feed documents. The bits turns off when the document passes under the head after the reader stops trying to feed documents. The bit also turns off because of a jam condition between the separator and the read head.
- When a hopper runout occurs, the line remains conditioned for about 850 milliseconds after the last document is fed (until the sorter is stopped line becomes active).
- I/O ready indicates that the reader is selected for a read operation. When the document reaches the read head, the system must be ready to start receiving data from the reader.

Byte	Bit	Name	Indicates	Reset By
0	0	Reserved		
0	1	End request	Used by CE for diagnostic programming	CE action
0	2	Interrupt pending	Reader requires program action	Next SIO issued
0	3	I/O attention	Normal operator intervention is required at the reader because one of these conditions occurred: – Full stacker – Empty hopper – The reader has stopped with the feed light on – Document feeding has been stopped because the stop key has been pressed – No validity check and readout key is pressed A jam that occurs in the separator area is indicated as an empty hopper condition. Error conditions (feed jam, transport, interlock, and stacker command) inhibit an I/O attention indication.	Correcting the condition that stopped the reader and pressing the START key
0	4	Data transfer register parity error	The SIOC detected a parity error in at least one byte of data passing through the data transfer register for transmission to the read data field in main storage.	
0	5	No-op	The last instruction issued to the SIOC is rejected because the reader is not capable of performing the operation specified by the instruction.	
0	6	Length count register overflow	The number of bytes specified for transfer to the read data field, after being read by the reader, were transferred.	
0	7	I/O ready	The reader was selected for a read operation.	

Figure 11-4. 1255/1270 Status Byte

Program Notes for 3881

- Figure 11-5 describes the 3881 status bits and their meanings.

- Equipment check indicates a detected 3881 read head or microprocessor failure. If the SIOC presents equipment check as the result of a read command, ignore any data transferred to the processing unit during that command execution.

- Output record ready bit indicates that th 3881 has completed reading all the marks from a form passing under the read heads, translated the marks into EBCDIC data bytes or mark image bytes, and stored them (along with special status bytes that relate to the data just read) in the 3881 read buffer. The output record ready bit is reset when the last byte is transmitted to the system by a read command. The program should test this bit after each read command is executed to determine whether another read command must be issued to read more data from the buffer to the system.

- The program can obtain additional status information from the record descriptor word, which is part of each 3881 output record. The record descriptor word (Figure 11-2) contains 2 bytes of status information that indicate if the form was selected by the 3881, if the record contains a serial number, etc. For details about the 3881 output record and the record descriptor word, see *IBM 3881 Optical Mark Reader Models 1 and 2 Reference Manual and Operator's Guide,* GA21-9143.

Byte	Bit	Name	Indicates	Reset By
0	0	Reserved		
0	1	End request	CE uses this bit for diagnostic programs.	CE action
0	2	Interrupt pending	The 3881 requires program action.	
0	3	Not used by 3881		
0	4	Data transfer register parity check	The SIOC detected a parity error in at least one byte of data passing through the data transfer register for transmission from the 3881 read buffer to the read data field in CPU storage.	
0	5	No-op	The last instruction issued to the SIOC was rejected because the 3881 is not capable of performing the operation specified by the instrucion.	
0	6	Length count register overflow	The number of bytes specified for transfer to the read data field from the 3881 read buffer has been transferred.	
0	7	I/O ready	The 3881 has been selected for a read operation.	

Figure 11-5. 3881 Status Bytes

IBM 5471 Printer-Keyboard

The IBM 5471 Printer-Keyboard is mounted on the system tabletop with a forms stand located on the floor behind it. The keyboard and the printer are not physically linked; that is, pressing a key does not automatically cause a character to be printed on the printer. The printer and keyboard are housed together and the printer motor is used to restore the keyboard.

5471 PRINTER CHARACTERISTICS

The printer prints 10 characters per inch on a 12.5-inch writing line. The entire 64-character system character set can be printed except for minus zero. The printer signals the attachment when it begins an operation and when it ends the operation. Printing or spacing requires about 64.5 milliseconds per character. Carrier return operates at about 15 inches per second.

5471 KEYBOARD CHARACTERISTICS

The keyboard can generate the system character set except for minus zero. The keys are interlocked to prevent pressing two keys simultaneously. In addition to the system graphics set, special codes are generated. Automatic restoration of the keyboard after operation of a graphic, SHIFT, or RETURN key requires about 64.5 milliseconds.

5471 ATTACHMENT CHARACTERISTICS

5471 Keyboard Attachment

Before an operation can be performed on the printer-keyboard, interrupts must be enabled by the processing unit. Three different interrupt conditions can be enabled or disabled:

- Interrupts caused by pressing the request key
- Interrupts caused by pressing any key other than the request key
- Interrupts caused by the completion of a printer operation

All of these interrupts are independent of each other and any combination can be enabled at any one time. If more than one is enabled, the stored program must test the interrupt-pending sense bits to determine the cause of the interrupt. If the keyboard and printer interrupt simultaneously, the keyboard should be serviced first.

When the printer-keyboard requires service, the attachment generates an interrupt-pending condition. If that interrupt has been enabled by the processing unit, a program interrupt request is generated on interrupt level 1.

Pressing the request key causes the request-key-interrupt-pending sense bit to be turned on. This status remains on until the processing unit issues a reset keyboard interrupt command. If the request-key interrupt is enabled or becomes enabled before the interrupt-pending is reset, a program interrupt request is generated. If the interrupt-pending status is generated while interrupts are disabled, or if the enable interrupt and reset interrupt commands are issued simultaneously on the same SIO instruction, the interrupt is lost.

The END key and the CANCEL key are treated exactly like the REQUEST key with the following exceptions:

- An end-key-or-cancel key-interrupt-pending status bit is generated.
- The interrupts from keys other than the REQUEST key must be enabled to allow the generation of the program interrupt request.

In the case of the graphic keys and the RETURN key, a graphic-key-or-return-key-interrupt-pending sense bit is generated. Interrupts from keys other than the REQUEST key must be enabled to allow the graphic keys or the RETURN key to cause a program interrupt request. Note that when interrupts from keys other than the REQUEST key are enabled, the end-key-or-cancel-key-interrupt-pending and the graphic-key-or-return-key-interrupt-pending sense bits must be tested to determine the cause of an interrupt request.

Graphic characters are handled in the following manner:

- The graphic keys are encoded to a keyboard code character with parity.
- The attachment translates the keyboard code character into the appropriate card code character.
- The card code character is translated to EBCDIC by the translator circuits in the I/O channel when the character is sent to storage.

If a parity error occurs in either the input or the output of the keyboard-to-card-code translator, a corresponding sense bit is also stored.

5471 Printer Attachment

In order to print a character, the character must be loaded into a print character buffer in the attachment by an LIO instruction. During loading, the I/O channel translates the character from EBCDIC to card code, and the attachment then translates the character from card code to a tilt-rotate code used to position the print element. The print mechanism is checked for correct shift at this time. The card-code-to-tilt-rotate translator includes a check bit, and if incorrect parity is detected on the output of the translator, a translator check sense bit is generated. If the character loaded into the print-character buffer is outside the printable character set, the tilt-rotate code for a space is established, and a nonprintable-character sense bit is generated.

After the character has been loaded in the print-character buffer, the stored program must issue a start print command through an SIO instruction. If the print mechanism is in the correct shift, a print cycle starts. If the print mechanism is not in the correct shift, a shift cycle precedes the print cycle.

Carrier return is controlled by the stored program. The carrier is initiated by an SIO instruction that designates carrier return. Carrier return moves the carrier to the left margin and advances the forms. A carrier return issued when the carrier is at the left margin only advances the forms.

Start print and start carrier return commands cause the printer to become busy. SIO and LIO instructions to the printer are not accepted while the device is busy. If a start print or start carrier return command is issued while the printer is interrupting without a simultaneous reset interrupt command, the printer interrupt request can not be reset until after the device becomes not busy. (The printer becomes not busy when the operation is complete.) The transition from busy to not busy generates a printer-interrupt-pending status. If printer interrupts are enabled, or if they become enabled before a reset printer interrupt command is given, a program interrupt request is generated.

The printer mechanism is checked against the nominal time required for each operation. If the timing is wrong, a printer-malfunction sense bit is generated, and a bit specifying the conditions that caused the printer malfunction bit (feedback too late, extra cycle, or cycle too long) is generated. These bits are turned off by a SNS instruction that detects them.

The printer contains contacts that detect the approach of the carrier to the end of the print line within 4 to 6 character spaces and the end of the form within 4 to 6 lines. These contacts set sense bits in the attachment.

To aid the CE in servicing the printer-keyboard, the following signals are made available as sense bits:

- The states of the three enable interrupt latches
- The input to the keyboard-code-to-card-code translator
- The output from the card-code-to-tilt-rotate translator and the printer-uppercase-mode switch
- The states and signals from the strobe and feedback contacts

5471 START I/O (SIO)

Op Code (hex)	Q-Byte (binary)	R-Byte (binary)
Byte 1	Byte 2	Byte 3
F3	0001 x 000	xxxx xxxx

DA M N Control Code

Bits	If M-code is 0	If M-code is 1
0	Spare; should be 0	Start print
1	Spare; should be 0	Start carrier return
2	Request pending indicator 1 = on 0 = off	Spare; should be 0
3	Proceed indicator 1 = on 0 = off	Spare; should be 0
4	Spare; should be 0	Spare; should be 0
5	Request key interrupts 1 = enable 0 = disable	Printer interrupt 1 = enable 0 = disable
6	Other key interrupts 1 = enable 0 = disable	Spare; should be 0
7	Reset request key and other key interrupts	Reset printer interrupt

N-code must be 000; any other N-code causes a processor check.

0 specifies the keyboard.
1 specifies the printer.

0001 specifies the 5471 printer-keyboard as the addressed device.

F3 specifies a start I/O operation. F as the first hex character in the op code specifies a command-type instruction (that is, an instruction with no operand addressing).

Operation

The printer or keyboard as specified by the M-code performs the operation specified by the control code.

Program Note

Always set spare control bits to 0.

5471 LOAD I/O (LIO)

Op Code (hex)	Q-Byte (binary)	Operand Address	
Byte 1	**Byte 2**	**Byte 3**	**Byte 4**
31	0001 1 000	Operand 1 address	
71	0001 1 000	Op 1 disp from XR1	
B1	0001 1 000	Op 1 disp from XR2	

DA M N

N-code must be 000; any other N-code causes a processor check.

1 specifies the printer.

0001 specifies the 5471 as the addressed device.

31, 71, or B1 specifies a load I/O operation. The first hex character in the op code specifies the type of operand addressing for the instruction.

Operation

This operation transfers the high-order (leftmost or lower numbered) byte of the 2-byte operand into the print-character buffer in the printer-keyboard attachment. The operand is addressed by its low-order (rightmost or higher numbered) storage position.

Program Note

You must place the character to be printed in the print-character buffer with this instruction *before* issuing the start-print command. However, if the same character is to be printed several times in succession, it need be loaded only once.

5471 SENSE I/O (SNS)

Op Code (hex)	Q-Byte (binary)	Operand Address	
Byte 1	Byte 2	Byte 3	Byte 4
30	0001 x 000	Operand 1 address	
70	0001 x 000	Op 1 disp from XR1	
B0	0001 x 000	Op 1 disp from XR2	

DA M N

N-Code	Sensed Unit
001	Sense bytes 0 and 1[1]
011	Sense bytes 2 and 3[1]

Any N-code not shown is invalid and causes a processor check.

0 specifies the keyboard.
1 specifies the printer.

0001 specifies the 5471 as the unit to be sensed.

30, 70, or B0 specifies a sense I/O operation. The first hex character in the op code specifies the type of operand addressing for the instruction.

[1] Odd-numbered sense bytes are transferred to the rightmost (higher numbered) position of the operand. Even-numbered sense bytes are transferred to the leftmost position of the operand.

Operation

The 5471 attachment transfers to the operand address 2 bytes of sense information as specified by the N-code. Status bits are defined in Figure 12-1. The operand is addressed by its low-order (rightmost or higher numbered) storage position.

TEST I/O AND BRANCH AND ADVANCE PROGRAM LEVEL INSTRUCTION

These instructions are not used by the printer-keyboard. An attempt to use either of these instructions with the printer-keyboard device address results in a processor check.

Byte	Bit	Indicates for Keyboard	Indicates for Printer
0	0	Not used	Printer enable
0	1	Not used	5.24 milliseconds[1]
0	2	Card code B	2.68 seconds[1]
0	3	Card code A	Cycle latch[1]
0	4	Card code 8	Reserved
0	5	Card code 4	Feedback too late
0	6	Card code 2	Extra cycle
0	7	Card code 1	Cycle too long
1	0	Request key interrupt pending	Printer interrupt pending
1	1	End or cancel interrupt pending	Reserved
1	2	Cancel key	Nonprintable character
1	3	End key	Printer busy
1	4	Return or data key interrupt pending	End of line
1	5	Return key	End of form
1	6	Keyboard translator check	Printer translator check
1	7	Keyboard check	Printer malfunction
Note: Sense bytes 2 and 3 are for CE use.			
2	0	Keyboard upper case mode switch	Printer upper case mode switch
2	1	Keyboard data reed switch P	No print
2	2	Keyboard data reed switch B	Tilt-rotate code T2
2	3	Keyboard data reed switch A	Tilt-rotate code T1
2	4	Keyboard data reed switch 8	Tilt-rotate code R5
2	5	Keyboard data reed switch 4	Tilt-rotate code R2A
2	6	Keyboard data reed switch 2	Tilt-rotate code R2
2	7	Keyboard data reed switch 1	Tilt-rotate code R1
3	0	Request key enabled	Lower shift required
3	1	Other key enabled	Upper shift required
3	2	Strobe switch	Reserved
3	3	Strobe switch sampled	Feedback switch
3	4	Request or end or cancel key	Feedback switch sampled
3	5	Request or end or cancel key sampled	Long FN switch
3	6	Keyboard shifting	Long FN switch sampled
3	7	Reserved	CE sense bit
[1] Used for diagnostic purposes			

Figure 12-1. 5471 Printer-Keyboard Sense Bytes

IBM 5475 Data Entry Keyboard

The IBM 5475 Data Entry Keyboard is comprised of a keyboard, control panel, covers, cables, and a set of attachment circuitry. The keyboard is designed to look and operate as much as possible like the keyboard for the IBM 5496 Data Recorder. Data recording and data verifying can be performed by using the data entry keyboard in conjunction with the card handling capabilities of the MFCU. The functions of data recording and verifying are available when the keyboard and system are used together under control of the data recording and data verifying programs that are available from IBM. The data recording and data verifying functions can be performed when the system is not needed for data processing programs. With the Dual Program Feature installed, data recording and data verifying functions also can be performed while the system is being used for data processing programs.

Communication between the processing unit and the keyboard is on an interrupt basis. Each time the keyboard needs the processing unit in order to perform its function, it must signal with interrupt. The keyboard is assigned interrupt level 1, which is next to last in interrupt priority.

Pressing the keys on the keyboard causes interrupts to occur. Certain switches on the control panel also cause interrupts to occur. Each key or switch also causes some status condition or data byte to appear in status bytes in the attachment. These status conditions and data bytes can be sampled by the processing unit to determine what procedure is to be followed after each key is pressed.

5475 KEYS AND SWITCHES

The keys on the keyboard are of two types, as are the switches on the control panel. Keys can be either latched keys (which require specific action to restore them from their operated position) or momentary contact keys (which return to the nonoperated position as soon as pressure is released). Switches are either two-position toggle switches that retain the position to which they are moved, or momentary-contact toggle switches that return to the non-operated position as soon as they are released.

5475 PROGRAM Switch is used by the data recording and data verifying programs to designate that data recording or data verifying is to be done under program control. This is a two-position toggle switch that causes an interrupt request and sets a status bit each time it is transferred from one position to the other.

5475 PROGRAM LOAD Switch is used to indicate that a program card for the data recording or data verifying program is to be loaded from the MFCU. It is a momentary-contact toggle switch that causes an interrupt request only when the program switch is on. The status bit set by this switch is set only as long as the switch is held transferred. The sense I/O instruction must be executed before the operator releases the switch in order for this bit to be sensed.

5475 RECORD ERASE Switch is used by the data recording and data verifying programs to erase the data in the record that is currently being entered into storage. This momentary-contact toggle switch causes an interrupt request each time it is operated and maintains the status bit only so long as the switch is operated.

5475 AUTO RECORD RELEASE Switch is used by the data recording and data verifying programs to determine if the card is to be processed as soon as the manual entries are completed. This two-position toggle switch causes an interrupt request each time it is moved from one position to the other and its state is available as a status bit.

5475 AUTO SKIP/DUP Switch is used by the data recording and data verifying programs to determine if fields coded as automatic operation fields in the program card are to be treated as automatic fields. This two-position toggle switch causes an interrupt request and changes a status bit each time the switch is transferred.

5475 PRINT Switch determines whether the data being keyed is to be printed on the card after being punched. This two-position toggle switch does not cause an interrupt request but does control a status bit.

5475 Function Keys

Function keys are momentary-contact type and are not interlocked from each other or from the rest of the keys. When an interrupting function key is pressed, the data keys are mechanically locked out and no other function key can generate an interrupt until the first key has been released. If multiple function keys are held down when a sense I/O operation occurs, all the bits will be recorded in the sense byte. If the sense I/O operation is not performed before the function key is released, the function key bit is not recorded in the sense byte. The attachment treats the momentary-contact toggle switches (program load switch and record erase switch) as interrupting function keys and these two rules also apply.

5475 UPPER SHIFT Key conditions the attachment logic to encode upper shift characters. This key does not generate interrupt requests and does not set a status bit.

5475 LOWER SHIFT Key conditions the attachment logic to encode lower shift characters. This key does not generate interrupt requests but sets a status bit if it is held down during a sense I/O operation.

5475 MULTI-PUNCH Key is pressed to place the keyboard in upper shift and, if the processing unit has unlocked the keyboard, causes each data key that is pressed to be restored. The encoded characters associated with the data keys that are pressed are logically ORed in the attachment. When the MULTI-PUNCH key is released, an interrupt request is generated. If no data key is pressed while the MULTI-PUNCH key is pressed, no interrupt request is generated by releasing the MULTI-PUNCH key.

5475 PROGRAM 1 Key is used to select the program 1 area as the location of the program control card. This key generates an interrupt request only if the program switch is on. If the key is held down during a sense I/O operation, a sense bit is set.

5475 PROGRAM 2 Key is used to select the program 2 control card area. It operates in the same as the PROGRAM 1 key.

5475 RELEASE Key signals the end of manual entries on the card. This key generates an interrupt request and sets a status bit when a sense I/O operation is performed while it is pressed.

5475 FIELD ERASE Key is used by the data recording and data verifying programs to signal that the last manually entered field is to be erased. The key causes an interrupt request and sets a status bit if the key is held down until a sense I/O operation is performed to detect it.

5475 ERROR RESET Key is used to reset program-detected errors. It results in an interrupt request and conditions a status bit if the sense I/O operation is performed while the key is down.

5475 READ Key is used by the data recording and data verifying programs to cause a card to be read into a data area of storage. This key causes an interrupt request and must be held down until the sense I/O operation occurs in order to record the status bit.

5475 SKIP Key is used by the data recording and data verifying programs to indicate that the remainder of the field is to be skipped. If the PROGRAM switch is on, this key generates a single interrupt request each time it is pressed. If the PROGRAM switch is off, interrupt requests are generated each 1/10 second as long as the key is held down. The down position on the key is recorded in the sense byte if the key is pressed when the sense I/O operation is performed.

5475 DUP Key is used by the data recording program and the data verifying program to signal that the remainder of the field is to be duplicated from the preceding card. If the PROGRAM switch is on, this key generates a single interrupt request each time it is pressed. If the PROGRAM switch is off, interrupt requests are generated each 1/10 second as long as the key is held down. The down position of the key is recorded in the sense byte if the key is held pressed when the sense I/O operation is performed.

5475 RIGHT ADJUST Key is used by the data recording and data verifying programs to signal that the data in the field is to be moved to the right end of the field and the remaining left end field positions filled with blanks. Pressing this key generates an interrupt request only if the PROGRAM switch is on. The RIGHT ADJUST key causes a sense bit if the sense I/O operation is performed while the key is pressed.

5475 Data Keys

The dual SHIFT keys, plus the space bar, are designated as data keys. These keys generate the 63 characters shown on the keyboard plus the code for blank. The data keys are latched type keys and are mechanically interlocked to prevent pressing more than one key at a time, but a second key can be pressed while the first key is held down if a keyboard restore cycle occurs after the first key is pressed. The attachment generates an interrupt request each time a data key is pressed and the character generated by that key is presented as a sense byte. The data keys must be restored by the processing unit, and a second key cannot be pressed until the processing unit restores the first.

The character generated by pressing a data key depends on the shift of the keyboard. The shift is determined in the following manner:

1. If the PROGRAM switch is off, the keyboard is in lower shift.

2. If the PROGRAM switch is on, the keyboard is in upper shift unless the program control card specifies otherwise or the lower shift key is pressed.

3. With the PROGRAM switch on, the program control card can specify numeric mode (through a start I/O instruction to the keyboard) for the next entry. In the numeric mode, pressing any key other than 0 through 9 or the space bar causes the attachment to turn on the invalid character bit in the sense byte. For the 0 through 9 keys, the attachment operates in upper shift.

4. Any of the preceding shift conditions can be manually overridden by pressing the SHIFT keys or the MULTI-PUNCH key. Manually determined shift states are effective only for as long as the determining key is held down.

Once a data key is pressed, all other data keys are mechanically locked out. The restoring of the data keys is controlled by the program through the use of the start I/O instruction. One bit in the start I/O instruction control code causes the key which has been pressed to be restored, but leaves the keyboard in such a state that all the data keys are locked out. Another control code bit causes the keyboard to be unlocked so that data keys can operate. If the control code contains both bits, a complete restore cycle occurs. However, the following caution should be noted:

CAUTION
If a start I/O initiates a complete restore cycle and another start I/O that does not have the unlock-keys control code bit on is issued before the key that was depressed has been restored (about 15 to 20 milliseconds), the data keys will all be locked out.

5475 Indicators

Indicators are provided on the control panel section of the keyboard to indicate the next column in which data will be entered, an error condition has occurred, and the program level control card that is in effect at the moment.

5475 Column Indicators are controlled by the program. The indicators are made up of segments that can be lighted in various combinations to produce the Arabic numeral characters. Other characters can be produced, but are not likely to be used for column identification.

5475 Error Indicator is lighted under program control. It is controlled by a bit in the start I/O control code.

5475 Program 1 and Program 2 Indicators are lighted under program control to indicate the program control card level that is in use.

5475 PROGRAMMING CONSIDERATIONS

The following rules must be observed in programming for the data entry keyboard:

1. A start I/O instruction to enable interrupt level 1 must be issued before the keyboard can be used.
2. A start I/O instruction must be issued to unlock the keyboard before the data keys are operable.
3. A sense I/O operation is necessary to obtain data from the keyboard.
4. The processing unit must issue an instruction to restore the data keys.
5. The last instruction in the interrupt routine must be a start I/O instruction to reset the interrupt.

5475 START I/O (SIO)

Op Code (hex)	Q-Byte (binary)	R-Byte (binary)
Byte 1	Byte 2	Byte 3
F3	0001 0 000	xxxx xxxx

DA M N Control Code

Bit	Condition
0	*Programmed numeric mode:* When this bit is on, the attachment is placed in numeric shift. When this bit is off and the PROGRAM switch is on, the attachment is placed in upper shift unless the lower shift bit is on. If both the lower shift bit and this bit are on, the attachment is placed in numeric shift. If a data key causes an interrupt, this bit must not be changed before the data is sensed.
1	*Programmed lower shift:* When this bit is on, the attachment is placed in lower shift. When this bit is off and the PROGRAM switch is on, the attachment is placed in upper shift unless the numeric shift bit is on. If both the numeric shift bit and this bit are on, the attachment is placed in numeric shift. If a data key causes an interrupt, this bit must not be changed before the data is sensed.
2	*Error indicator:* When this bit is on, the error indicator is lighted. When this bit is off, the error indicator is turned off.
3	Spare.
4	*Restore data key:* This bit causes the mechanism that restores and locks the data keys to operate. It leaves the keyboard in such condition that all the data keys are prevented from operating.
5	*Unlock data key:* This bit releases the restore and lock mechanism for the data keys. If both the restore and unlock bits are used in the same instruction, the attachment first restores the data keys then unlocks them.
6	*Enable/disable interrupt:* When this bit is on, the attachment is allowed to interrupt the program in progress in the processing unit. When this bit is off, the attachment is blocked from interrupting the processing unit.
7	*Reset interrupt:* When this bit is on, it resets the interrupt condition in the attachment and allows the processing unit to return to the program it was processing when the interrupt occurred.

N-code must be 000; any other N-code causes a processor check.

M-code must be 0.

0001 specifies the 5475 as the addressed device.

F3 specifies a start I/O operation. F as the first hex character in the op code specifies a command-type instruction (that is, an instruction with no operand addressing).

Operation

This instruction sets the conditions specified in the control code into the attachment.

5475 LOAD I/O (LIO)

Op Code (hex)	Q-Byte (binary)	Operand Address	
Byte 1	Byte 2	Byte 3	Byte 4
31	0001 0 000	Operand 1 address	
71	0001 0 000	Op 1 disp from XR1	
B1	0001 0 000	Op 1 disp from XR2	

DA M N

N-code must be 000; any other N-code causes a processor check.

M-code must be 0.

0001 specifies the 5475 as the addressed device.

31, 71, or B1 specifies a load I/O operation. The first hex character in the op code specifies the type of operand addressing for the instruction.

Operation

The 2 byte field located at the operand address controls the segments of the column indicator and turns on or off the program 1 and program 2 indicators. The operand is addressed by its rightmost byte. The rightmost byte controls the segments of the units position of the column indicator and the program 2 indicator. The high-order byte controls the tens position of the column indicator and the program 1 indicator. The segments of the column indicator are designated by letters as shown in Figure 12-2. Each bit in the bytes controls one segment or a program indicator. When a bit is on, the indicator or segment turns on. When a bit is off, the indicator or segment turns off. The bit assignments for the segments and indicators are:

Bit	Lights
0	Segment E
1	Segment D
2	Segment F
3	Segment C
4	Segment B
5	Segment G
6	Segment A
7	Program indicator

The hexadecimal digits to be placed in each byte to obtain the decimal digits are:

Decimal	Hexadecimal
0	EE
1	24
2	BA
3	B6
4	74
5	D6
6	DE
7	A4
8	FE
9	F6

If a program indicator is to be controlled, 1 must be added to the low-order hexadecimal digit for the appropriate byte.

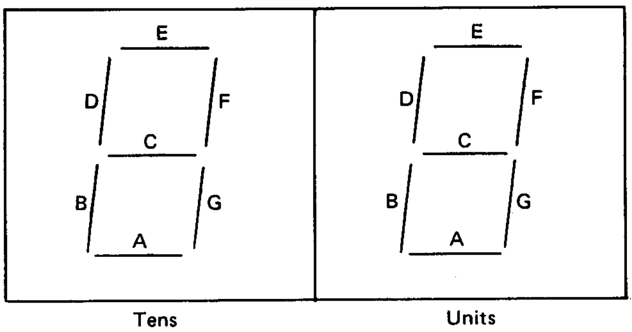


Figure 12-2. Column Indicator Arrangement

5475 SENSE I/O (SNS)

Op Code (hex)	Q-Byte (binary)	Operand Address	
Byte 1	Byte 2	Byte 3	Byte 4
30	0001 x 000	Operand 1 address	
70	0001 x 000	Op 1 disp from XR1	
B0	0001 x 000	Op 1 disp from XR2	

DA M N

N-Code **Sensed Unit**[1]

N-Code	Sensed Unit
001	Sense bytes 0 and 1
010	Sense bytes 2 and 3
011	Sense bytes 4 and 5

Any N-code not shown is invalid and causes a processor check.

M-code must be 0.

0001 specifies the 5475 as the unit to be sensed.

30. 70, or B0 specifies a sense I/O operation. The first hex character in the op code specifies the type of operand addressing for the instruction.

[1]Odd-numbered sense bytes are transferred to the rightmost (higher numbered) position of the operand. Even-numbered sense bytes are transferred to the leftmost position of the operand.

Operation

The 5475 attachment transfers to the operand address 2 bytes of sense information as specified by the N-code. Status bits are defined in Figure 12-3. The operand is addressed by its low-order (rightmost or higher numbered) storage position.

Program Notes

The reserved bits are always on 1. The function key interrupt, multi-punch interrupt, and data key interrupt bits indicate the cause of any program interrupt generated by the keyboard attachment. (Function key interrupt is turned on by the interrupting toggle switches as well as by the interrupting function keys.) The following programming requirements exist with regard to these 3 interrupt bits:

1. One and only one of the 3 bits should be on any time the keyboard attachment generates a program interrupt request. If none or more than one of these bits is on, a malfunction has occurred. In this case, the keyboard should be locked and the operator forced to try again. This can occur if a data key is pressed and, in servicing the interrupt, the program locks the keyboard by failing to restore that key. If a function key interrupt is generated after this operation, both the function key interrupt (correct) and the data key interrupt (from the unrestored data key) will be on.

2. If an interrupt is generated by changing the state of the PROGRAM switch, the AUTO SKIP/DUP switch, or the AUTO RECORD RELEASE switch, the function key interrupt bit is automatically reset 3.3 milliseconds after the interrupt is generated.

3. If the interrupt request is a function key interrupt, the data character should be ignored.

TEST I/O AND BRANCH AND ADVANCE PROGRAM LEVEL

These instructions are not used with the data entry keyboard. An attempt to execute either instruction with the data entry keyboard device address (0001) results in a processor check stop with an invalid Q-byte indication.

Byte	Bit	Meaning	Byte	Bit	Meaning
0	All	This byte is a data character	3	0	Auto skip/dup switch on
1	0	Print switch on		1	Record erase switch operated
	1	Reserved		2	Reserved
	2	Lower shift key pressed		3	Program switch on
	3	Invalid character detected		4	Skip key pressed
	4	Reserved		5	Dup key pressed
	5	Multi-punch interrupt		6	Auto record release switch on
	6	Reserved		7	Function key interrupt
	7	Data key interrupt	4	All	This byte is not used; all bits will be 0
2	0	Program 1 key pressed	5	0	Keyboard interrupts enabled
	1	Program 2 key pressed		1	Any function key pressed
	2	Program load switch operated		2	Bail forward contacts
	3	Release key pressed		3	Unlock keyboard signal
	4	Field erase key pressed		4	Bail forward trigger
	5	Error reset key pressed		5	Toggle switch on
	6	Read key pressed		6	Any data key
	7	Right adjust key pressed		7	CE sense bit

Figure 12-3. 5475 Sense Bytes

The System/3 Model 15 display station consists of:

- An IBM 3277 Model 1 Display Station—a CRT
- A 78-key Operator Console Keyboard (Feature 4632)

The CRT/keyboard is the operator/system communication device for Model 15, and is required on every system. It is attached directly to the processing unit and is located on the system tabletop. The CRT/keyboard can be used for inquiry, limited output, limited key entry of data, and operator/program interaction.

The CRT can display 480 characters: 12 lines of 40 characters each. It supports a 64-character set (36 alphameric characters, 27 special characters, and a blank). The keyboard is a movable 78-key EBCDIC keyboard with 45 alphameric keys: 21 control keys and 12 program function keys. As each character is keyed, the 3277 automatically sends the character to the CRT where it is displayed. After a meaningful unit (up to 480 characters) of data is entered onto the display from the keyboard, the operator can inspect the displayed data and correct the data, if necessary, from the keyboard. The entire process, to this point, requires no processing unit activity. Then, reacting to an operator-initiated interrupt, the system program can issue a read command to store the displayed data.

The CRT also allows the system to display error messages, register contents, and stored data (including instructions) under program control. If hard copy must be produced to record data (in addition to displaying the data), or if the system requires a second printer for the application being run, an IBM 3284 Model 1 Printer can be attached to the system. The 3284 uses a matrix print head and pin feed platen and has a rated speed of 40 characters per second. It prints the same set of 64 EBCDIC characters (including the blank) that is supported by the display. The maximum print line is 132 characters, but shorter lines can be printed if the data is formatted by system programming.

3277/3284 FUNCTIONAL DESCRIPTION

The 3277 attachment serves as an in-transit buffer storage and control unit between the attached I/O devices and the I/O channel. Attachment functions are performed by a microprocessor under control of a permanently defined stored program. (This program must be loaded after powering up and attachment check and must not be changed.) Data transfer between the 480-byte attachment buffer and main storage is by cycle steals via the I/O channel. Data is transferred serially by byte up to a maximum of 480 bytes on a write or read operation.

Data transfer between the attachment and the CRT or printer is serially by bit. Parity is checked at the receiving end. Data transfer between the attachment and the device is always 480 characters, including attribute control characters.

The device address is hex 10 for both the 3277 display station and the 3284 printer.

This adapter uses one processing unit local storage register as a main storage data address register. This register is called the 3277/3284 data address register (CRTAR). Op-end interrupts (end of data transfers) and unit interrupts (3277 attention keys; 3284, printing complete) are both handled on interrupt level 1.

The 3277 attachment performs control and I/O channel interfacing functions with the display and printer, and provides capability to:

- Continuously poll the CRT/keyboard terminal and respond to operator program action initiated from the keyboard.
- Identify and respond to commands issued by the processing unit.
- Interface with the system I/O channel and both the 3277 and 3284 interface channels.
- Perform data serialization/deserialization and error detection functions.
- Provide in-transit message storage and control on data transfer operations.
- Assemble attachment, display, and printer status for error recovery and diagnostic evaluation by the processing unit.

3277/3284 Microprogram

The microprograms are located in the attachment read/write control storage. They consist of a series of micro instructions arranged in routines to process SIO service requests. The following operations are controlled by the microprogram:

- Decode and execute SIO (nonimmediate) instructions or instruction decoding and execution.
- I/O interrupt requests
- I/O cycle steal requests
- Cursor address handling
- Message buffer addressing
- Length count incrementing
- Status and diagnostic byte generation
- Line polling
- Transmission error checking
- Line selection
- Preparation of 13-bit words for transmitting
- Analysis of 13-bit words during receiving
- Data translation

The processing unit program starts line operations by issuing an SIO (nonimmediate) instruction. If the instruction is accepted, an SIO service request is set on. The microprogram decodes the stored Q-byte N-code during an SIO service routine and selects the proper device (the 3277 or 3284). It then decodes the R-byte and performs the specified function.

3277/3284 High Density Buffer (HDB)

The HDB is a 32-byte storage area that the microprocessor can read into or write from.

If the attachment is disabled, the low half of the HDB can be written and ready by LIO and SNS commands to ensure proper parity after power up provide diagnostic capability.

If the attachment is enabled, the high half of the HDB can be written and read by LIO and SNS commands (this is the operating mode).

3277/3284 Interrupts

Interrupts for the 3277/3284 are in two forms: op-end interrupts and unit interrupts.

Op-end interrupts occur when the attachment completes the current SIO command.

Unit interrupts occur (1) when the 3284 unit completes printing, (2) when a program access key is pressed on the 3277 causing an attention condition, or (3) if error conditions develop.

Both interrupts are presented serially to the system as they occur. All information regarding the interrupt is constructed in the interrupt condition register by the microprogram, which then sets the 3277/3284 interrupt request latch.

An interrupt at level 1 occurs when interrupts are enabled. A TIO interrupt pending indicates the presence of the 3277/3284 interrupt request. A SNS command to the interrupt condition register provides information regarding the interrupt. An SIO immediate instruction with bit-6 of the R-byte (control code) on resets the interrupt request and causes the program to exit the interrupt level.

Should the microprogram need to generate additional interrupts before the current 3277/3284 interrupt is processed, it accumulates the information in the high density buffer (HDB). When the previous interrupt has been processed, the microprogram presents the interrupt.

Up to 17 interrupts can be stored sequentially in the HDB; then the attachment hangs and appears busy until an interrupt is processed.

Note: When an interrupt occurs, the program should issue a TIO instruction to determine that the interrupt was not caused by an attachment check. Issuing a SNS instruction with an M-code of 0 and an N-code of 110 accesses the contents of the interrupt condition register. The bits stored in this register define the cause of the interrupt.

3277/3284 Polling Operation

The microcontroller continuously polls the 3277 and 3284 for device status when it is not executing commands. It stores device status in the attachment for the following reasons:

- Indications of error conditions (these can be used for error recovery)
- Interrupt requests initiated by the operator when keyboard interrupts are allowed
- Device busy status
- Interrupt request (op-end) after a 3277 read or write operation ends or after a 3284 print operation ends

The processing unit program can examine the device status by sensing the interrupt condition register.

3277/3284 Interrupt Condition Register

The microcontroller loads status bits into the interrupt condition register during 3277 and 3284 operations. These bits define the reason the attachment posted an op-end or unit interrupt request.

When an interrupt occurs, the processing unit program should first issue a TIO for attachment check. If there is none, the program should sense the interrupt condition register. The byte entering the position specified by the operand address (the higher numbered storage position) contains interrupt condition register bits 8 through F. This byte is sense byte 1. The byte entering the position specified by the operand address minus 1 contains interrupt condition register bits 0 through 7, and is identified as sense byte 2.

Figure 13-1 defines the interrupt condition sense bits.

3277/3284 Data Handling

Data handling involving the use of buffers is specified during write or read SIO commands. The number of bytes transferred to or from the I/O channel may be specified by the count in the length count register. If the starting address plus length count exceeds 480, an op-end interrupt occurs with the program error bit (sense bit C in the interrupt condition register) on.

The number of 13-bit words transferred to or from the display or matrix printer is 480. Upon completion of this data transfer sequence, the microprogram polls the device for status to determine if any errors occurred.

3277/3284 Message Buffer

The attachment has a 512-byte message buffer that holds data being transferred between main storage and the selected device (the 3277 or the 3284). Only 480 bytes hold data and control characters that constitute the message being sent to or from the selected unit. The other 32 positions are used to store interrupt data.

3277/3284 Data Storage

Both the display and matrix printer have unit buffers that can store 480 characters.

Alphameric and attribute characters, along with cursor information, can be stored in each device buffer location.

Each buffer location in the 3277 attachment contains 8 bits plus parity. The stored data represents the printable characters and attributes. Defined hex values specify the attribute characters.

3277/3284 Attribute Characters

Attribute characters describe the characteristics of the field that follows them in storage and having the following properties:

- Attribute characters occupy displayable character locations in storage, but are not displayed or printed.
- No alphameric character can be stored simultaneously in a character location occupied by an attribute character.
- Attribute characters are protected; that is, they cannot be replaced by alphameric characters from the keyboard.
- Attribute characters can be entered under program control.
- A buffer is defined as being formatted when it contains at least one attribute character.
- Attributes used on the System/3 Model 15 are protect and unprotect. The protect attribute is represented by a hex A0 and unprotect by a hex 80. These codes appear in the data stream.
- Attribute character bit functions are as follows:

Bit 0 Always a 1
Bit 1 Always a 0
Bit 2 0 = Unprotected
1 = Protected (Note 1)
Bit 3 0 = Alphameric
1 = Numeric (Note 1)
Bit 4 and 5
00 = Displayed, not detectable (Note 2)
01 = Displayed, detectable (Note 2)
10 = Intensified and detectable (Note 2)
11 = Not displayed, nonprint, not detectable (Note 2),
Bit 6 Reserved
Bit 7 Modified data tag (MDT)—Not used
When data within a field defined by attributes is changed from the keyboard the MDT bit of the attribute that defines the start of the *next* field is set to 1.

Notes:
1. If bits 2 and 3 equal binary 11, a skip to the next field takes place.
2. Detectable and not detectable refer to the selector pen feature. This feature is not supported by Model 15.

- The attribute character function is part of the 3277 and 3284. Care must be taken when loading the units. If random information placed in the units contains attribute characters, unusual and unexpected results may occur.

3277 Cursor Address Register (CURAR)

The 3277 unit buffer holds 10 bits per word: 8 hex bits plus parity plus 1 cursor bit. Because the attachment message buffer is only 9 bits (8 hex bits plus parity), it is necessary to retain cursor position as an address (rather than as a bit within a word). Valid cursor addresses are hex 0000 through 01DF.

During read, when the cursor bit is detected, the MBAR contents are saved. When all of the message has been received, the microprogram moves the CURAR into the HDB for sensing by the program.

During write, the program can provide the cursor address to the attachment by an LIO. The microprogram loads this address into the CURAR. When the CURAR equals the MBAR, a cursor bit is inserted in the outgoing data stream. If the program does not perform an LIO to provide a new cursor address, the old cursor address is used, thus leaving the cursor in its original position. However, if a total write instruction is issued and no LIO is issued, the cursor is placed at the first position.

3284 Printer Variable Line Format

When this format is specified, the buffered printer honors new line (NL) and end of message (EM) printer control characters, if they are present in the printable field. Printout of the data in the buffer begins at character position 0 and continues until the last character position of the buffer is printed, or until an EM character is encountered.

When an EM character is encountered, the printing operation is terminated. None of the data following the EM character in the buffer is printed. All character positions in the buffer, with the exception of NL and EM characters, result in a printer character.

Attribute characters, null characters, and space characters are printed as spaces.

When an NL character is encountered in the buffer, the printer performs a line advance/carriage return operation. If no NL character is encountered before the printer prints in the last available character position on the line (determined by platen size), the printer performs a line advance/carriage return operation and continues printing.

The NL and EM characters that appear in the data stream as hex 35 and 39 characters are displayed as printable characters 5 and 9. The printer prints them as a 5 and 9 when not in a variable line format mode of operation.

3277 Cursor Display and Control

The cursor is a special symbol on the display to indicate the character position of the next alphameric character entered from the keyboard. One bit of storage in each character location in the 3277 buffer is provided for storing cursor information. Thus, the cursor may be simultaneously stored and displayed in any one attribute or alphameric character storage location in the buffer.

By the Unit

A display station power on operation causes the cursor to be generated, stored, and displayed in buffer location 0. This same operation resets the associated device buffer to all null characters.

The cursor can be repositioned at the display by keyboard operations. The printer cannot control cursor movement.

By the System

The cursor can be repositioned under program control when a write SIO is executed. The cursor register must be loaded to the desired position. During execution of the write command, when the word counter is equal to the cursor register on data transfer to the display, the cursor bit is set.

The cursor can be set with data or attribute characters.

3277 OPERATOR CONSOLE FUNCTIONS

3277 Keyboard Availability

The ability of an operator to perform a keyboard operation depends upon:

- The type of operation
- Conditions at the keyboard
- Whether or not an SIO is in process with the display

3277 Keyboard Disable

When a keyboard is disabled, the INPUT INHIBITED light is on and the only keyboard operations honored are specified RESET key operations.

A keyboard can become disabled in the following ways:

1. An SIO operation is in process with the display to which the keyboard is attached.
2. Operation of any program access key.
3. Operation of any alphameric key, the PA1 key, the CANCEL key, the ERASE EOF key, or the DELETE key when the keyboard is inhibited.
4. A parity error or cursor check is detected in the device buffer.
5. An SIO nonimmediate issued to the CRT with R-byte, bit 7 on.

Condition 3 requires operation of the RESET key to unlock the keyboard. All other causes of a disabled keyboard require program action.

3277 Keyboard Inhibit

Only cursor-positioning keys, program access keys, and operator function keys (with the exception of PA1, CANCEL, ERASE EOF, and DELETE) may be successfully operated when the keyboard is inhibited under the following conditions:

1. The cursor is located, via data entry, in a protected field or in a buffer position occupied by an attribute character.

2. The display is in insert mode, and no null character exists at the character location occupied by the cursor, or between the cursor location and the end of that field.

3277 Keyboard Interaction

The display station is capable of honoring only one input data source at a time. As a result, operations resulting in transfer or modification of the buffer (attachment-initiated buffer transfers or keyboard operations) are mutually exclusive.

3277 Keying Rate

The keyboard can be keyed at a rate of 25 characters per second.

3277 Cursor-Positioning Controls

Cursor-positioning keys permit rapid positioning of cursor to any character position on the display image. All cursor-positioning keys can operate and perform their function when the keyboard is inhibited but not disabled.

3277 Character-Oriented Keys

The up, down, left, right, and backspace keys are exclusively cursor positioning, and in no way alter attribute or alphameric characters stored in the 3277 unit buffer.

The cursor may be moved in protected or unprotected alphameric character and attribute character positions using cursor-positioning keys. Al these keys can cause the cursor to wrap. Horizontal wrap always involves a vertical movement; the cursor repositions to the next row of characters. Vertical wrap, due to operation of up or down keys, involves no horizontal movement; the cursor stays in the same character column.

3277 Field-Oriented Keys

1. TAB—Pressing this key causes the cursor to move to the first character position of the next unprotected data field. This key can cause the cursor to wrap. In an unformatted display, the cursor is repositioned to character position 0.

   The tab key has automatic repeat capability, at a rate of six operations per second.

2. BACK TAB—Pressing this key, when the cursor is located in the attribute character or the first alphameric character position of an unprotected data field or in any character position of a protected data field, causes the cursor to move to the first alphameric character position of the first preceding unprotected data field. Pressing this key when the cursor is located in any alphameric character location of an unprotected data field other than the first location, causes the cursor to move to the first alphameric character position of that field. This key can cause the cursor to wrap. In an unformatted display, the cursor is repositioned to position 0.

3. NEW LINE—Pressing this key causes the cursor to move to the first unprotected character position of the next line containing unprotected characters. If the display has no unprotected data fields, the cursor moves to character position 0. If the display is unformatted, the cursor moves to the first character position of the next line. This key can cause the cursor to wrap.

   The new line key has automatic repeat capability.

3277 Shift Keys

SHIFT

When the cursor is located in an unprotected field, simultaneously pressing a data key and a SHIFT key enters the upper character of a data key into the 3277 unit buffer.

LOCK

The LOCK key, when pressed, locks the keyboard in a shifted condition and enters the upper of the two character symbols shown on the key. After pressing LOCK, a SHIFT key must be pressed to restore the keyboard to its usual (unshifted) operating condition.

3277 Operator Function Keys

Operator function keys, unlike program access keys, do not generate an AID (attention ID) character or an I/O pending (interrupt) condition.

ERASE EOF (Erase End of Field)

If the cursor is located in an alphameric character position in an unprotected data field, pressing this key clears to nulls the character position occupied by the cursor and all remaining character positions in that field.

While clearing all character positions from the cursor position to the end of the field, the operation can wrap from the end of the last line on the display to the beginning of the top line. The cursor does not move as a result of operating this key.

Operation of this key when the cursor is located in an attribute character position or within a protected data field, disables the keyboard; no character positions are cleared and the cursor is not moved.

ERASE INPUT

Pressing this key clears all unprotected character positions to nulls, and repositions the cursor to the first unprotected character position on the display.

In an unformatted buffer, the entire buffer is cleared to nulls and the cursor is repositioned to position 0.

In a buffer with no unprotected data fields, no character positions are cleared and the cursor is repositioned to position 0.

INSERT MODE

Pressing INSERT MODE lights the insert mode indicator and places the keyboard controls in an insert mode of operation. Subsequently pressing alphameric keys while in insert mode results in the operations described in the following paragraphs.

If the cursor is located in an unprotected data field, with a null character in the character position occupied by the cursor or a null character in any character location in the data field beyond the cursor, pressing an alphameric key enters that alphameric character in the character position occupied by the cursor. The unprotected character, formerly occupying the cursor location, and all remaining characters within the unprotected data field (except for null characters or characters to the right of the null characters), are shifted one character position to the right. If the character occupying the cursor location at the time of the insert operation is a null, no character shifting occurs.

If a gap in alphameric characters exists at or beyond the cursor location (due to the presence of null characters), alphameric characters are shifted into the gap (with each character entry) until all null characters are overwritten. Subsequently pressing alphameric keys shifts characters beyond the former null characters gap to the right. After all null characters at or beyond the cursor location in the field are overwritten, or if no null characters exist, pressing an alphameric key disables the keyboard. Attribute characters remain in their fixed character positions, and are not shifted as part of the insert operation.

If more than one row of characters is contained within the field, a character occupying the last character position in the row is shifted into the first character position of the next row.

If the cursor is located in a protected field, pressing an alphameric key in insert mode disables the keyboard.

The insert mode of operation affects only keyboard alphameric entries. All keyboard function keys perform in the usual manner. If an I/O operation is honored during the time the keyboard controls are in insert mode, the I/O operation is performed in the usual manner.

Pressing RESET in insert mode, returns the keyboard controls to normal operation and turns off the INSERT MODE light.

DELETE

If the cursor is located in an alphameric character position in an unprotected field, pressing this key deletes the character from the character position occupied by the cursor. The cursor does not move. All remaining characters in the unprotected field, to the right of the cursor and on the same row, shift one character position to the left. Vacated character positions at the end of the row are filled with nulls. If the unprotected field encompasses more than one row, characters in rows other than the row occupied by the cursor are not affected.

Operation of this key is identical in insert mode.

If the cursor is located in a protected field, pressing this key disables the keyboard.

RESET

Pressing RESET enables the keyboard and turns off the INPUT INHIBITED light if the keyboard was disabled because of one of the following conditions:

1. Operation of any alphameric key, the PA1 key, the CANCEL key, the ERASE EOF key, or the DELETE key while the keyboard was inhibited.

2. Operation of a program access key. If the program subsequently issued a start I/O instruction specifying control only with unlock keyboard, the RESET key will not be needed. If the program issued a start I/O instruction that specified read or write with unlock keyboard, pressing the RESET key enables the keyboard.

RESET is normally used to recover from a keyboard operation that resulted in a disabled keyboard. For example, a keyboard disabled due to an attempted character entry into a protected field can be restored to normal operation by pressing RESET. The operator can then tab out of the protected field.

3277 Program Access Keys

The program access keys alert the program and/or solicit program action as a result of setting I/O pending; by an attention status indication.

Operation of any program access key disables the keyboard, lights the INPUT INHIBITED indicator, and turns off the SYSTEM AVAILABLE indicator.

The keyboard may be restored (unlocked) with a subsequent control only SIO command, or by operation of the RESET key.

CLEAR

Pressing this key clears the entire 3277 unit buffer to nulls. The display hardware establishes an unformatted display and moves the cursor to character position 0.

Pressing CLEAR sets I/O interrupt pending and the clear AID.

ENTER

Pressing this key sets I/O interrupt pending and the enter AID.

PA1 (Program Attention 1) and CANCEL

Pressing these keys sets I/O interrupt pending and the associated AID.

TEST REQ (Test Request)

Pressing this key sets I/O interrupt pending and the test request AID.

PF (Program Function)

Pressing any of these keys (PF1 through PF12) sets I/O interrupt pending and the appropriate PF key AID.

3277 Alphameric Character Keys

Keyboard entry of alphameric character code into the 3277 unit buffer occurs at the cursor location, provided the cursor is located in an alphameric character position within an unprotected data field. In an attribute character position, it locks the keyboard.

If the attribute character terminating a field defines the following field as an alphameric, and either an unprotected or protected data field, the cursor skips the attribute character and moves to the first alphameric character position in the following data field. This cursor skip occurs whether the following field is defined as protected or unprotected.

If one or more successive attribute characters follow the attribute character terminating that field; that is, if no alphameric character follows the attribute character terminating that field, each attribute character is examined as noted previously. The cursor moves to the first alphameric character location following the successive attribute character group.

Alphameric character keys include the complete 63-character EBCDIC set, including space. The space bar and the underscore/minus key have automatic repeat capability.

The character displayed by the underscore character is a single horizontal line that is the bottom of the character matrix. The display system does not perform an underscore function; that is, it does not simultaneously display the underscore character and any additional character, in a single character position.

3277 SYSTEM AVAILABLE Indicator

This indicator is turned on by any command to the 3277. It is turned off when an interrupt generating key (PF key) is pressed.

INITIALIZING THE 3277/3284 ADAPTER

The 3277/3284 adapter uses a microcontroller to perform many of its functions. The microcontroller operates under control of a microprogram that is provided by IBM; this microprogram must *never* be altered. Before operating the adapter after a power down condition and after an attachment check condition, the adapter must be initialized and the microprogram must be loaded into control (microcontroller) storage.

If you are using IBM programming support, all of these functions are performed during the IPL procedure. Otherwise, the following procedure initializes the adapter, loads the microprogram, and enables both the adapter and the microcontroller:

If you are using IBM programming support, all of these functions are performed during the IPL procedure. Otherwise, loading CE deck FFF, then CE deck 143, then CE deck FC0 initializes the adapter, loads the microprogram, and enables both the adapter and the microcontroller.

3277/3284 TEST I/O AND BRANCH (TIO)

Op Code (hex)	Q-Byte (binary)	Operand Address	
Byte 1	Byte 2	Byte 3	Byte 4
C1	0001 1 xxx	Operand 1 address	
D1	0001 1 xxx	Op 1 disp from XR1	
E1	0001 1 xxx	Op 1 disp from XR2	

DA M N

N-Code	Condition
000	Attachment not ready
001	Interrupt pending
010	HDB/external parity check[1]
011	Control storage parity check[1]
100	Storage address parity check[1]
101	Attachment check
110	Storage write data parity check[1]
111	Attachment busy

M-code must be 1; an M-code of 0 results in a no-op instruction.

Hex 1 specifies the 3277/3284 attachment as the unit to be tested.

C1, D1, or E1 specifies a test I/O and branch operation. The first hex character in the op code specifies the type of operand addressing for the instruction.

[1] Attachment check is also indicated. This condition further defines the cause of the attachment check.

Operation

The processing unit tests the 3277/3284 attachment for the condition specified by the N-code. If the tested condition exists, the program branches to the instruction stored at the operand address. If the tested condition does not exist, the processing unit accesses the next sequential instruction. (If the branch occurs, the processing unit stores the branch-to address—from the operand address—in the instruction address register and the next sequential instruction address in the address recall register. If the branch does not occur, the processing unit stores the branch to address in the address recall register and leaves the contents of the instruction address register unchanged.)

Resulting Condition Register Setting

This instruction does not affect the condition register.

Program Notes

- The *attachment not ready* condition is caused by (1) attachment disabled, (2) microcontroller disabled, or (3) attachment check.

- The *interrupt pending* condition occurs when (1) an op-end interrupt pending condition or (2) a unit interrupt pending condition. Op-end interrupt pending occurs when data transfer is completed during execution of a 3277 or 3284 SIO instruction. Unit interrupt pending occurs when any program attention key is pressed, or when the 3284 goes from busy to not busy (printing completes).

- The *attachment check* condition occurs when the attachment detects a parity error. An attachment check causes an interrupt request, but does not give a positive response to a TIO instruction testing the *any device interrupt pending* condition.

- The *attachment busy* condition applies from the time an SIO that specifies data transfer is issued until the attachment completes the execution of the instruction. After a delay of about 10 cycles, an interrupt request occurs.

- Whenever an interrupt request occurs, the program should issue a TIO for attachment check. If there is none, the program should issue an SNS interrupt condition register to determine the cause of the interrupt.

3277/3284 ADVANCE PROGRAM LEVEL (APL)— MODEL 10 MODE ONLY

Op Code (hex)	Q-Byte (binary)	R-Byte (binary)
Byte 1	Byte 2	Byte 3
F1	0001 1 xxx	0000 0000

DA M N

The R-byte is not used in an APL instruction.

N-Code	Condition
000	Attachment not ready
001	Any device interrupt pending
010	HDB/external parity check[1]
011	Control storage parity check[1]
100	Storage address parity check[1]
101	Attachment check
110	Storage write data parity check[1]
111	Attachment busy

The M-code must be 1; an M-code of 0 results in a no-op instruction.

Hex 1 specifies the 3277/3284 attachment as the unit to be tested.

F1 specifies an advance program level operation. F as the first hex character in the op code, specifies a command-type instruction (that is, an instruction with no operand addressing).

[1] Attachment check is also indicated. This condition further defines the cause of the attachment check.

Operation

The processing unit tests the 3277/3284 attachment for the unique condition specified by the N-code. If the tested condition exists, the program loops on the APL instruction until the condition no longer exists, then advances to the next sequential instruction. If no tested condition exists, the processing unit immediately accesses the next sequential instruction.

Program Notes

The program notes for the 3277/3284 TIO instruction also apply to this instruction.

3277/3284 LOAD I/O (LIO)

Op Code (hex)	Q-Byte (binary)	Operand Address	
Byte 1	Byte 2	Byte 3	Byte 4
31	0001 x xxx	Operand 1 address	
71	0001 x xxx	Op 1 disp from XR1	
B1	0001 x xxx	Op 1 disp from XR2	

DA M N

M	N-Code	To Be Loaded
0	000	3277/3284 attachment message buffer address register[1]
	010	3277/3284 count register[1]
	101	Cursor address register[1]
1	00x	Code for an instruction used for no function except loading the microprogram into the microcontroller after supplying power to the system. The programmer should never use this code. If used at any time except in the IBM-supplied microprogram, attachment functioning will be impaired.
	011	3277/3284 data address register (CRTAR).

Any N-code not shown is invalid and causes:
- Program check if interrupt level 7 is enabled
- Processor check if interrupt level 7 is not enabled

Hex 1 specifies the 3277/3284 attachment as the unit whose registers are to be loaded.

31, 71, or B1 specifies a load I/O operation. The first hex character in the op code specifies the type of operand addressing for the instruction.

[1] All other N-codes with M-code = 0 are used by the microprogram for attachment control only and, although accepted by the attachment as valid codes, should *not* be used by the programmer.

Operation

The processing unit loads the contents of the 2-byte field specified by the operand into the register specified by the N-code.

Program Notes

- The message buffer address register should be loaded before a partial read or write of the attachment buffer to provide a message buffer starting address for the data transfer to or from main storage. It is only used for an SIO nonimmediate read or write command with bit 3 of the control code set to 1.

- Load the byte count register with the number of bytes to be transferred to or from main storage before issuing an SIO nonimmediate read or write instruction with bit 3 of the control code set to 1.

- When data is transferred to the display buffer, a cursor bit always accompanies the data. During display of the data, the cursor is displayed at the position holding the cursor bit:
  - If a new address is not loaded into the cursor address register before the program issues a partial write SIO command, the cursor will remain at the position it occupied prior to the write command.
  - If a new address is loaded into the cursor address register before the program issues a total write command, the cursor is placed at the first buffer position.
  - If a new address is loaded into the cursor address register before either a partial or a total write SIO, the cursor is placed at the new address.

- The 3277/3284 data address register (CRTAR) must contain the address of the appropriate 3277 or 3284 data field in main storage before the program issues a read or write SIO command. This address specifies the first (lowest numbered) position of the field (1) into which data will be read from the display unit, (2) from which data will be written to the display, or (3) from which data will be written to the printer.

- The microcontroller stored program must be loaded after each power-up sequence and before using either the 3277 or 3284. To do this, perform the initialization procedure specified by IBM, using the IBM-supplied microprogram.

3277/3284 SENSE I/O (SNS)

Op Code (hex)	Q-Byte (binary)	Operand Address	
Byte 1	Byte 2	Byte 3	Byte 4
30	0001 x xxx	Operand 1 address	
70	0001 x xxx	Op 1 disp from XR1	
B0	0001 x xxx	Op 1 disp from XR2	

DA M N

M	N-Code	Register
0	101	Cursor address register
0	110	Interrupt condition register
0	111	Lines (3277 and 3284) busy register.

All other N-codes are accepted if M = 0; however, these have no meaning to the processing unit program and should not be issued.

M	N-Code	Register
1	00x	These codes are accepted by the attachment but have no meaning to the processing unit and should not be used.
1	010	Invalid
1	011	3277/3284 data address register
1	1xx	Invalid

Any invalid N-code causes:
Program check if interrupt 7 is enabled
Processor check if interrupt 7 is not enabled

Hex 1 specifies the 3277/3284 attachment as the unit to be sensed.

30, 70, or B0 specifies a sense I/O operation. The first hex character in the op code specifies the type of operand addressing to be used for the instruction.

Operation

The processing unit moves the data from the register specified by the M-code and N-code into the 2-byte main storage field specified by the operand address. The field is addressed by its higher numbered (low-order or rightmost) storage position. The first byte moved (called byte 1) enters the position addressed by the operand address. The second byte moved enters the operand address minus 1 storage position. Figure 13-1 describes the sense byte meanings.

Program Notes

- The cursor address register can be sensed after a read command to acquire the position of the cursor bit in the record just read. If the operator has used the keyboard, the address in the cursor address register will not indicate the present location of the cursor on the display.
- Sensing the interrupt condition register provides the necessary information to describe the current interrupt request. If there is no current interrupt request, the contents of this register is not significant.
- Sensing the lines busy register indicates which lines are busy. Byte 2 is stored at the operand address -1 position, byte 1 is stored at the operand address:
  - Byte 2, bit 0 = 1 indicates the 3277 is busy.
  - Byte 2, bit 1 = 1 indicates the 3284 is busy.
  - Byte 2, bits 2-7 have no meaning.
  - Byte 1, bits 0-7 have no meaning.

Never sense the lines busy register while the attachment is busy.

Bit	Sense Byte 2 Meaning If: Tested Bit Is Off	Tested Bit Is On	Suggested Testing Sequence
0	Op-end interrupt	Unit interrupt	1
1, 2	Not used (will always be off)	Not used (will always be off)	NA
3	Applies to 3277	Applies to 3284	2
4	No error or check occurred	Error or check occurred	3
5	No transmit check occurred	Transmit check occurred	–
6	No receive check occurred	Receive check occurred	–
7	No device check occurred	Device check occurred	–

Bit	Sense Byte 1 Meaning If Tested Bit Is On: With O-End Interrupt	With 3277 Unit Interrupt	With 3284 Unit Interrupt
0	Control check occurred	Control check occurred	Control check occurred
1	Not used	Not used	Equipment check occurred
2	Printer-did-not-go-busy	Operator pressed a program access key	Not used
3	Not ready/no-response	Operator pressed one of these keys: TEST REQ, CLEAR, or PF1 through PF12.	3284-went-not-ready
4	Program error occurred	Operator pressed one of these keys: PA1, CLEAR, CANCEL, ENTER, or PF8 through PF12	Not used
5	Print-busy	Operator pressed one of these keys: PA1, CLEAR, CANCEL, ENTER, PF4, PF5, PF6, PF7, or PF12.	Not used
6	Not used	Operator pressed one of these keys: CANCEL, PF2, PF3, PF6, PF7, PF10, or PF11	Not used
7	Not used	Operator pressed one of these keys: CLEAR, ENTER, PF1, PF3, PF5, PF7, PF9, or PF11	3284-went-not-busy

Note: Resetting the interrupt described by the information in the interrupt condition register resets all bits in that register and allows the information for the next sequential interrupt (if any) to be moved into the interrupt condition register. This new information can now be sensed.

Figure 13-1. 3277/3284 Sense Bytes (Interrupt Condition Register Bytes)

3277/3284 START I/O (SIO)

Op Code (hex)	Q-Byte (binary)	R-Byte (binary)
Byte 1	Byte 2	Byte 3
F3	0001 x xxx	xxxx xxxx

```
          DA    M  N       Control Code
                   N-Code       Bits
                                0123  4567 (meaning with M-code = 1)
                1¹ 000          xxx0  00xx
                                            0 = disable interrupt
                                            1 = enable interrupt
                                           1 = reset interrupt request
                                      Should be 000 (diagnostic bits)
                                1x0 = disable microcontroller²
                                1x1 = enable microcontroller²
                                10x = disable attachment²
                                11x = enable attachment²

                0  000  selects 0123  4567 (meaning with M-code = 0)
                        3277    xxxx  00xx
                                        With 3277 selected:
                                        67 (these bits provide keyboard control)
                                        xx
                                        0 = unlock keyboard
                                        1 = lock keyboard
                                        1 = sound keyboard alarm
                0  001  selects         With 3284 selected:
                        3284            67 (these bits provide printer format control)
                                          00 = NL and EM (new line and end-of-message) characters determine format³
                                          01 = print with 40 characters per line
                                          10 = print with 64 characters per line
                                          11 = print with 80 characters per line
                                      Must be 00
                                  1 = use buffer address register and count register contents (loaded by LIO)
                                000 = control only
                                010 = read (with control)
                                100 = write (with control)
                                110 = erase unprotected (with control)

          Hex 1 specifies the 3277/3284 attachment as the unit being controlled.
```

F3 specifes a start I/O operation. F, as the first hex character in the op code, specifies a command-type instruction (that is, an instruction with no operand addressing).

---

[1] An M-code of 1 causes an immediate operation without a resulting op-end interrupt. An M-code of 0 causes a non-immediate operation that usually requires data transfer time; with this type of instruction, an op-end interrupt occurs at end of data transfer. For print operations, a unit interrupt occurs upon completion of printing.

[2] To reset any interrupts that may be buffered from another program, the attachment and microcontroller should be disabled, then immediately enabled. With this exception, the microcontroller should never be disabled except by the IBM-provided 3277/3284 microprogram load routine. Therefore, bit 2 of the control code (R-byte) of an immediate SIO should usually be set to 1 for application programming.

[3] NL (new line) character causes the form to advance one line space. The hex code for NL is 35. EM (end-of-message) character terminates the print operation. An EM character in any position except position 1 also causes the form to advance one line space. The hex code for EM is 39.

Operation, General

The processing unit performs the function specified by the control code on the device or attachment specified by the M- and N-codes. (The SIO instruction is accepted by the attachment unless an I/O attention condition or busy condition exists.)

3277 Read Operation

After accepting a read command, the attachment transfers the entire 480-byte content of the 3277 unit buffer to the attachment buffer, then examines bit 3 of the control code (R-byte) to determine whether the read operation is a partial read (bit 3 on) or a total read (bit 3 off). For a total read operation, the attachment now sends the entire contents of the attachment message buffer to the main storage data field designated by the address in the data address register, requesting an o-end interrupt when the last byte from the buffer has been transferred to storage. For a partial read operation, the attachment examines the contents of the attachment buffer address register for the location of the first byte to be transferred from the attachment message buffer to the processing unit. Starting with this location, the attachment transfers the number of bytes specified by the count register (previously loaded into the register by an LIO instruction) from consecutive message buffer positions to the main storage data field designated by the address in the data address register. The attachment requests an op-end interrupt when the last byte to be transferred is transferred to the processing unit.

Note: The program can sense the cursor address register to determine the location of the cursor on the display. The keyboard is locked during read operations. If control code bit 7 is on, the keyboard remains locked at the end of the operation; if bit 7 is off, the keyboard unlocks when the read operation ends. However, if the keyboard was disabled by operation of a program access key before the read operation started, the keyboard must be unlocked by issuing a control only SIO, or the operator must reset the keyboard by pressing RESET.

3277/3284 Write Operation

After accepting a write command, the attachment examines bit 3 of the control code (R-byte) to determine whether a partial write operation or a total write operation is to be performed.

If bit 3 is off, a total write operation is to be performed; and the processing unit sends 480 consecutive bytes of data from main storage, starting at the position specified by the data address register, to the 480-byte attachment message buffer. After the buffer is filled, the attachment sends the entire contents of the message buffer to the unit buffer of the selected device (3277 or 3284). If the 3277 is selected to write the data, the attachment inserts a cursor bit into the byte pattern for one of the characters being transferred. The first byte in the buffer contains the cursor bit unless the cursor address register has been loaded by an LIO. If the cursor address register has been loaded, the cursor bit accompanies the byte specified by the cursor address register. (The cursor is displayed at the location occupied by this bit.)

If bit 3 is on, a partial write operation is performed and the attachment reads the contents of the selected unit buffer (3277 or 3284) into the attachment message buffer. Then the processing unit transfers the number of characters specified by the count register from processing unit main storage to the attachment message buffer. The attachment stores these bytes in consecutive attachment message buffer positions, starting at the location specified by the message buffer address register. The new data replaces the old data in the position being loaded, but positions not receiving new data from the processing unit retain the data read from the unit buffer. After data transfer from the processing unit to the attachment is completed, the attachment moves the entire contents of the attachment buffer to the selected unit buffer. If the command specified the 3277, the contents of the 3277 unit buffer are not displayed. If the command specified the 3284, the attachment examines bits 6 and 7 of the command code (R-byte) to determine print line length and formats the printer accordingly. The 3284 then prints the contents of the 3284 unit buffer. The keyboard is locked during a write operation to the 3277; if the command code bit 7 is on, the keyboard remains locked when the operation ends. (Data written on the display is retained on the display by display circuits until the display is turned off or new data replaces it.) An op-end interrupt occurs when the attachment has completed the write operation. A unit interrupt occurs when any program function key or any attention key is pressed on the 3277 or when the 3284 goes from a busy state to a not-busy state (printing complete).

If the 3277 was specified, the cursor is displayed at the location addressed by the cursor address register. This is the previous cursor location unless a new address was loaded into the cursor address register before executing the write command.

3277 Erase Unprotected Operation

This is a 3277 operation. When the attachment accepts the command specifying this operation, it instructs the 3277 to perform the following functions:

- Clear all unprotected fields to null characters.
- Reset any pending attention indications.
- Set the cursor at the first character position of the first unprotected field.
- Unlock the keyboard.

The attachment then generates an op-end interrupt request.

3277/3284 ERROR DEFINITION AND RECOVERY

Transmit Check

A 480-character (word) data stream to the unit resulted in at least one parity check in the I/O device. The 3284 holds the erroneous record in its buffer but will not print it. The 3277 displays the erroneous record except that a null is displayed in place of the incorrect characters. If the information in the 3277 is not retransmitted correctly within approximately 20 milliseconds, a device check may also follow.

Recovery: Retry the write operation four times. If not successful, halt with a message or indication.

Note: If the transmit check occurs while doing a partial write, it will be necessary to reload the entire 3277 buffer.

Receive Check

At least one parity error was detected while receiving a 480 character record from the unit during a read instruction, or anytime while receiving status from the units. No data is changed in main storage if a receive check occurs.

Recovery: Retry the operation four times. If not successful, halt with a message or indication.

Device Check

The 3277 or 3284 detected either bad parity, more than one cursor, or no cursor in its storage buffer. This check is detected most often while polling. In the 3284, device checks that occur while polling are ignored unless waiting for a unit interrupt (waiting for the 3284 to go not busy after a successful write). In the 3277, the interrupt is always taken, but polling is suspended after the interrupt is given.

Recovery: The entire unit buffer must be reloaded. If the attempts to refresh the unit buffer fail after four retries, halt with a message or indication.

Printer Did Not Go Busy

A successful write to the 3284 and a subsequent start print (issued by the microprogram) did not result in the 3284 going busy, indicating that printing did not start.

Recovery: Retry the operation four times. If not successful, halt with a message or indication.

Not Ready/No Response

A start I/O to the addressed device did not result in any response to the initial poll, or the 3284 is not ready.

Recovery: Retry four times, then halt with a message or indication to check the following:

1. Is the unit attached and powered on?
2. Are the switches in their proper position and are forms properly inserted in the carriage in the 3284?

Program

A program error is caused by the following items:

1. A line other than 0 or 1 was selected.

2. Bits 2, 5, or 6 or the R-byte of the SIO were on.

3. During a partial read or write, the count plus the initial MBAR setting exceed the 480-character capacity of the unit; that is, MBAR (hex) + count (hex) is greater than 01E0 (hex).

   In the case of a partial read, main storage is changed for as many bytes until the error is detected. All other operations are suppressed.

4. MBAR or CURAR was set to a value greater than 01DF (hex).

Recovery: None, program must be corrected.

Control Check

A hardware failure in either the unit or attachment was detected. Results are unpredictable.

Recovery: Retry the operations four times. If not successful, halt with a message or indication.

Printer Busy

An SIO to the 3284 was initiated while it was busy.

Recovery: None required, no harm was done. To avoid getting these interrupts, wait for the 3284 unit interrupt which signals that the printer went not busy.

Printer Equipment Check

A failure occurred in the 3284. Erroneous information may be printed.

Recovery: Depends upon application. If permitted, retry four times and if not successful, halt with a message or indications.

Printer Went Not Ready

While printing, the 3284 went not ready before the operation was complete. This may be caused by two things:

1. A hang condition in the 3284 caused a printer malfunction.

2. Printing was suspended when the cover of the 3284 was raised during a print operation.

Recovery option 1: Wait 15 seconds and retry the operation. If not successful after four times, halt with a message or indication.

Recovery option 2: By using the switches in the 3284, the buffer contents may be printed as follows:

- Set mode switch to mode 2.
- Set test switch to print buffer.
- Activate the start print switch.
- Return the test switch back to the on line position before continuing.

If the above procedure is not used, the write SIO to the 3284 must be repeated.

CONSIDERATIONS FOR PROGRAMMING THE 3284 PRINTER USING IBM PROGRAMMING SUPPORT

There can be no new line (NL) characters embedded within the text. The NL characters may precede or follow the text.

If an end-of-message (EM) character immediately follows a new line character, bit 2 (hex 20) in the IOBPFL must be set on.

On partial writes, the data in the main storage buffer must be followed by an end-of-message character.

On partial writes, a displacement must be specified to indicate where data from the 3284 printer data field is to be placed in main storage (IOBPDP).

An end-of-message character in position 1 does not cause the printer to perform a line-space operation.

The new line (NL) character code is hex 35. The end-of-message (EM) character code is hex 39. These characters are significant only for a variable-line format write operation. They print as a 5 (NL graphic) and a 9 (EM graphic) on other format writes.

Items 1 through 5 are necessary to keep the current line position.

Chapter 14. IBM 3741 Data Station Models 1 and 2 and IBM 3741 Programmable Work Station Models 3 and 4

Either an IBM 3741 Data Station Model 1 or 2 or an IBM 3741 Programmable Work Station Model 3 or 4 can be directly attached to the IBM System/3. The attached 3741 can be used online to System/3 as a diskette input/output device or can be used offline to perform 3741 functions such as data entry and communications, and can be used (Models 3 and 4 only) as a programmable work station. An IBM System/3, using IBM programs, supports the 3741 Models 3 and 4 in data station mode only; the System/3 does not support the application control language for Models 3 and 4.

The publication, *IBM System/3 3741 Reference Manual* GC21-5113, which is available through the IBM branch office serving your locality, provides additional information about the operation and control of a directly attached 3741.

Data Transfer Rate

The 3741 directly attached to System/3 provides input/output rates of about 1500 records per minute when reading from the diskette, and about 1000 records per minute when writing to the diskette. These rates depend on the complexity of the application and the following assumptions:

- Records are transferred between the 3741 and a disk device (3340, 5444, or 5445).
- Diskette records contain 128 bytes, and the 3741 data is double buffered.
- The disk uses 1028-byte blocks (eight diskette records per disk block); and the data is double buffered.
- The system is dedicated (without spooling, multiprogramming, or dual programming).
- The operation is error-free, with no alternate tracks assigned on the disk or diskette.

Maximum Diskette Record Size

A diskette record may not contain more than 128 bytes.

Attachment to System

The 3741 is attached, via a signal cable, to an attachment feature in the processing unit.

Power

The 3741 has a power cord that receives power from a receptacle in the room. (The 3741 attachment feature is powered from the System/3.) Therefore, supplying power to the system does not supply power to the 3741. Also, there is no emergency power-off between the processing unit and the 3741.

Online Selection

A 3741 attached to the system can operate in either online mode or offline mode. In online mode, the 3741 keyboard is inoperative, except to take the 3741 offline. Data transfer between the 3741 and the system is always between the system and the diskette, never between the system and the keyboard/display screen.

Before the 3741 can be used as a system I/O device, the operator must bring up 3741 power, load a diskette into the 3741, and place the 3741 in online mode.

The system must check to determine that the 3741 is online before performing any 3741 I/O operations. If the operator has placed the 3741 offline, the program must detect this condition and halt.

REGISTERS AND PROGRAM-TESTABLE LINES

The following registers and testable lines are used to program 3741 I/O operations.

Data Transfer Register

A 9-bit data transfer register temporarily stores 1 byte of data (8 bits plus a parity bit) that is to be moved either direction between the diskette and main storage. Data transfer usually occurs on a cycle steal basis, but the contents of this register can be moved between the register and main storage with load I/O and sense I/O instructions. The system tests this register for correct parity, setting a sense bit if incorrect parity is encountered.

Length Count Register

Because data transfer occurs on a cycle-steal basis, the 3741 attachment must keep track of the number of bytes transferred. A length count register performs this function.

This register must be loaded before each I/O operation, using a load I/O instruction. The number to be loaded into the register depends on the number of bytes to be transferred. The number should be 255 minus the number of bytes to be transferred. For example, if you wish to transfer 128 bytes (the maximum length 3741 record), load hex 7F into the length count register. The 3741 signals when the last byte has been transferred from the diskette to the system.

Upon successful completion of a record transfer operation, the length count register should contain hex FF or a lower number. If the program loads a record length less than the physical record length, the system indicates an overflow that, when recognized by the 3741, places the 3741 in the offline mode.

If the program loads a record length greater than the physical record length, the length count register does not read hex FF at the end of data transfer.

The contents of the length count register and the overflow condition can be placed in storage for testing with a sense I/O instruction.

The attachment stores the overflow byte one position beyond the right end of the 3741 data field in main storage. For example, if the length count register specifies 96 bytes but the record contains 128 bytes, 97 bytes will be transferred.

I/O Transfer Lines

Although this set of signal lines from the 3741 to the attachment is not a register, the lines can be tested by a sense I/O instruction. The lines, which provide information about 3741 operations, status, and identification, can be used by the System/3 program for program decisions.

I/O transfer lines, their associated testable bits, and their meanings, are:

I/O Transfer Line	Associated Byte and Bit[1]	Meaning
1	Byte 1, bit 7	Not used
2	Byte 1, bit 6	Not used
3	Byte 1, bit 5	3741 attention required
4	Byte 1, bit 4	End-of-job out (3741 indicates end of job)
5	Byte 1, bit 3	End-of-record out (3741 indicates end of data transfer for a System/3 read instruction)
6	Byte 1, bit 2	Bus-in parity error (attachment detected a parity error in data received from System/3)
7	Byte 1, bit 1	End-of-data-set out (3741 indicates end of data set)
8	Byte 1, bit 0	Not used
9	Byte 2, bit 7	Read from attachment (3741 requests a System/3 write operation)
10	Byte 2, bit 6	Write to attachment (3741 requests a System/3 read operation)
11	Byte 2, bit 5	3741 online
12	Byte 2, bit 4	I/O cable attached
13	Byte 2, bit 3	I/O identification 1-bit (will be 0)
14	Byte 2, bit 2	I/O identification 2-bit (will be 0)
15	Byte 2, bit 1	I/O identification 4-bit (will be 1)
16	Byte 2, bit 0	I/O identification 8-bit (will be 0)

[1] Byte 1 is the low-order (rightmost, high address) byte. Byte 2 is the high-order (leftmost, low address) byte.

I/O Function Register

The program must load hex 4000 into the function register at the start of each job that uses the 3741. To load the function register, use the load I/O instruction.

3741 Data Address Register

This local storage register stores the address of the leftmost byte of the data field that is to be used for the next 3741 read or write operation. The register must be loaded before each read or write operation. The register can be sensed with a sense instruction.

3741 OPERATIONS

The 3741 reads data to the system from one sector of the diskette or writes data into one sector of the diskette under control of start I/O read or write instructions. Data is transferred sequentially, one byte at a time, in EBCDIC.

At the start of the job, the operator must manually place the 3741 in proper mode and online. Thereafter, the 3741 operates under System/3 program control.

Establishing Synchronism

The program must ensure that the 3741 is synchronized with the system before issuing each control SIO instruction. The following procedure can be used:

1. Sense the 3741 status byte for an interrupt pending condition.
2. Reset the interrupt pending latch by issuing an SIO instruction.

Initial Adapter Setup

The adapter must be set up once per job. Use the following procedure:

1. Reset the adapter.
2. Load the attachment function register with hex 4000.

Data Transfer

Use the following procedure to transfer data to and from the 3741:

1. Test with an SNS instruction to ensure that the 3741 is attached and online.
2. Test with a TIO instruction to ensure that the 3741 is ready and not busy.
3. Wait for the 3741 to bring up either the read, write, end-of-date, end-of-job, or 3741-attention-required indicator. On the Model 15, an op-end interrupt request accompanies the indication that occurs if op-end interrupt is enabled.
4. Respond to the 3741 request with an appropriate SIO instruction. Go to step 5 for a read or write indication.
5. Load the address of the first byte of data into the 3741 data address register.
6. Load 255 minus the number of bytes to be transferred into the length count register.
7. Issue a read or write SIO instruction.
8. Proceed to the error checking operation when a test for attachment busy fails.

Error Checking

The program should check for errors after each read or write SIO instruction has been executed. You can use the following procedure:

1. Sense the I/O transfer lines, the length count register, and the attachment status byte before issuing any subsequent SIO instruction.
2. Check for the following conditions:
   a. 3741 attached and online. (If the 3741 goes offline, no op-end interrupt is generated.) Halt if offline.
   b. No-op. If the last SIO instruction set the no-op bit, reissue the instruction.
   c. 3741 bus-in parity error (3741 detected). Send normal response to 3741 and resend record (starting at step 3 in data transfer procedure). Repeat this step, if necessary. If the parity error still persists after these 3 attempts to send the record, the 3741 sets the 3741 attention required indicator.
   d. 3741 bus-out parity error (attachment detected). Send 3741 bus-out parity error indication to 3741 and reread the record (starting at step 3 in the data transfer procedure). The 3741 makes as many as two more attempts to transmit the record without error. If a parity error persists, the 3741 sets the 3741 attention required indicator.
   e. 3741 attention required. Send normal response and halt.
   f. Length error. To determine this, check for length count register residual of hex FF. If the residual is not FF, a length count error has occurred and you should send a length count indication to the 3741. If no length count error occurred, send a normal response to the end-of-record indication. To ensure that you transfer no more data than you want, specify the number of characters to be transferred minus one in the length count register and check for end of record and length-count register residual of hex 00.

End-of-Record Processing

After each System/3 read operation, the program should perform the following procedure:

1. Check for end-of-job indication from 3741. If it is the end of the job, send a normal response to 3741 and halt.

2. Check for end-of-data indication from 3741. If there is an end-of-date indication, send normal response and process appropriately, then return to the data transfer operation.

At the end of data or end of job during a System/3 write operation, the program should:

1. Send end of data to the 3741 for end-of-data condition, then return to the data transfer operation, or

2. Send end of job to the 3741 for end-of-job condition.

3741 IPL Operation for Models 12, 15B, 15C, and 15D

The 3741 is used as the alternate program load device on Models 12, 15B, 15C, and 15D cardless systems.

Placing the program load selector switch in the ALTERNATE position and pressing the PROGRAM LOAD key on the processing unit panel starts the procedure. The attachment waits with the processing unit I/O ATTENTION light on until the operator places the 3741 online in the output-from-3741 mode. When the attachment recognizes that the 3741 is in the correct mode it reads the record available into storage at address 0000 in cycle steal mode. (This operation is similar to reading a record with a 3741 SIO read instruction.) When the complete record is in storage, the processing unit examines the contents of the IAR (which was set to 0000 when the PROGRAM LOAD key was pressed) and resumes normal operation. The record read into storage will cause reading of the rest of the IPL program from the 3741.

Record length of the IPL records must not exceed 128 bytes.

3741 START I/O (SIO)

Op Code (hex)	Q-Byte (binary)	R-Byte (binary)
Byte 1	Byte 2	Byte 3
F3	0100 0 xxx	xxxx xxxx

DA M N Control Code

N-Code	Bits 0123	4567	Function Specified
000[1]	0000	0001	Reset interrupt request (diagnostic except on Model 15)
	0000	0010	Enable interrupt request (diagnostic except on Model 15)
	0000	0100	Disable interrupt request (diagnostic except on Model 15)
	0000	1000	Reset 3741 attachment, removing 3741 attachment from busy state
	0001	0000	Set interrupt request (diagnostic except on Model 15)
001	0000	0000	Read
010	0000	0000	Write
011	0000	1000	Indicate normal response to 3741
	0001	0000	Indicate record-length error to 3741
	0001	0100	Indicate attachment (mode) error to 3741
	0011	0000	Indicate end of data set to 3741
	0101	0000	Indicate end of job to 3741
	1001	0000	Indicate 3741 bus out parity error to 3741[2]

Any N-code not shown is invalid and causes:
- Program check if interrupt level 7 is enabled on Model 15
- Processor check if interrupt level 7 is not enabled on Model 15
- Processor check on Models 8, 10, and 12

M-code is always 0 for the 3741.

Hex 4 specifies the 3741 as the device to be controlled.

F3 specifies a start I/O operation. F as the first hex character in the op code, specifies a command-type instruction (that is, an instruction without operand addressing).

[1] N-code 000 specifies interrupt control only. The interrupt control function can also be programmed with read and write instructions (N-codes of 001 and 010).

[2] Upon receipt of 3741 bus out parity error, the 3741 automatically retransmits the record in error. After 2 unsuccessful retransmissions, a 3741 attention required indication occurs.

Operation

The 3741 performs the function specified by the N-code and control code (R-byte).

3741 TEST I/O AND BRANCH (TIO)

Op Code (hex)	Q-Byte (binary)	Operand Address	
Byte 1	Byte 2	Byte 3	Byte 4
C1	0100 0 xxx	Operand 1 address	
D1	0100 0 xxx	Op 1 disp from XR1	
E1	0100 0 xxx	Op 1 disp from XR2	

DA M N

N-Code **Condition Tested**

000 3741 not ready/attachment check

010 Attachment busy

Any N-code not shown is invalid and causes:

Program check if interrupt level 7 is enabled on Model 15

Processor check if interrupt level 7 is not enabled on Model 15

Processor check on Models 8, 10, and 12

M-code for the 3741 is always 0.

DA-code for the 3741 is always hex 4.

C1, D1, or E1 specifies a test I/O and branch operation. The first hex character in the op code specifies the type of operand addressing to be used for the instruction.

Operation

The processing unit tests the conditions specified by the N-code. If the condition is present, the processing unit places the address from the instruction address register into the address recall register, then places the operand address from the instruction into the instruction address register, and then accesses that instruction. If the condition tested is not present, the processing unit places the operand address from the instruction in the address recall register and executes the address specified by the instruction address register (the next sequential address).

Program Notes

- The 3741 not-ready/attachment check TIO condition indicates that one or more of these conditions exist:

  1. No I/O device is attached to the 3741 attachment.

  2. The data transfer register has incorrect parity.

  3. A read or write SIO instruction addressed to the 3741 was accepted but was not executed (no-op condition). An op-end interrupt is never requested for a no-op condition on the 3741.

  4. The 3741 is not ready. The program must sense the status byte to determine which of these conditions exists.

- The attachment busy TIO condition indicates that the 3741 attachment is performing an operation.

CAUTION

Never loop on the TIO for ready condition. The program will not drop out of the loop.

3741 ADVANCE PROGRAM LEVEL (APL)

Op Code (hex)	Q-Byte (binary)	R-Byte (binary)
Byte 1	**Byte 2**	**Byte 3**
F1	0100 0 0xx	0000 0000

DA — M — N

R-byte is not used in an APL instruction.

N-Code	Condition Tested
000	3741 not ready/attachment check (do not use on Model 15)
010	Attachment busy

Any N-code not shown is invalid and causes:
- Program check if interrupt level 7 is enabled on Model 15
- Processor check if interrupt level 7 is not enabled on Model 15
- Processor check on Models 8, 10, and 12

M-code for the 3741 is always 0.

Device address of the 3741 is always hex 4.

F1 specifies an APL operation. F, as the first hex character in the op code, identifies a command-type instruction (that is, an instruction without operand addressing).

Operation

The processing unit tests the 3741 attachment for the condition specified by the N-code. If any tested condition exists, the program loops on the APL instruction until the condition no longer exists, then advances to the next sequential instruction. If no tested condition exists, the processing unit immediately accesses the next sequential instruction.

Program Note

If the DA- and M- portions of the Q-byte are binary 00000, the APL instruction is treated as a no-op command and the processing unit immediately accesses the next sequential instruction. For this unconditional no-op, the N-code should be binary 000.

3741 LOAD I/O (LIO)

Op Code (hex)	Q-Byte (binary)	Operand Address	
Byte 1	**Byte 2**	**Byte 3**	**Byte 4**
31	0100 0 xxx	Operand 1 address	
71	0100 0 xxx	Op 1 disp from XR1	
B1	0100 0 xxx	Op 1 disp from XR2	

DA M N

N-Code **To Be Loaded**

001 I/O function register (you must load hex 4000 into the register)
010 Length count register
100 3741 data address register
101 Data transfer register

Any N-code not shown is invalid and causes:
- Program check if interrupt level 7 is enabled on Model 15
- Processor check if interrupt level 7 is not enabled on Model 15
- Processor check on Models 8, 10, and 12

M-code is always 0 for 3741.

Hex 4 specifies the 3741 as the device whose registers are to be loaded.

31, 71, or B1 specifies a load I/O operation. The first hex character in the op code specifies the type of operand addressing to be used for the instruction.

Operation

The processing unit loads the 2 bytes of data contained in the operand into the register specified by the N-code. The operand is addressed by its low-order (higher numbered) storage position.

3741 SENSE I/O (SNS)

Op Code (hex)	Q-Byte (binary)	Operand Address	
Byte 1	**Byte 2**	**Byte 3**	**Byte 4**
30	0100 0 xxx	Operand 1 address	
70	0100 0 xxx	Op 1 disp from XR1	
B0	0100 0 xxx	Op 1 disp from XR2	

DA M N

N-Code **Data Source**

001 I/O function register
010 Length count register[1] and status byte
011 I/O transfer lines; these lines from the 3741 are bit significant as follows:

Low-Order Byte (operand address) Byte 1

Bit	*Meaning*
0	Not used
1	End of data[2]
2	3741 bus-in parity error[2]
3	End of record[2]
4	End of job[2]
5	3741 attention required[2]
6	Not used
7	Not used

High-Order Byte (operand address minus 1) Byte 2

Bit	*Meaning*
0-3	Must be 0100 (hex 4)
4	3741 attached
5	3741 online
6	Write to attachment request from 3741[2]
7	Read from attachment request from 3741[2]

100 3741 data address register
101 Data transfer register[1] and diagnostic byte

Any N-code not shown is invalid and causes:
- Program check, if interrupt level 7 is enabled on Model 15
- Processor check, if interrupt level 7 is not enabled on Model 15
- Processor check on Models 8, 10, and 12

M-code is always 0 for 3741.

Hex 4 specifies the 3741 as the device whose registers and lines are to be sensed.

Hex 30, 70, or B0 specifies a sense I/O operation. The first hex digit in the op code indicates the type of operand addressing for the instruction.

---

[1]This byte is stored at the operand address. The associated byte is stored at the operand address minus 1.
[2]This indication is accompanied by an op-end interrupt on Model 15 if interrupts are enabled for the 3741.

Operation

The processing unit moves data from the source specified by the N-code to the 2-byte field specified by the operand address.

Program Notes

- The operand is always addressed by the low-order (higher storage number) byte.

- This instruction is executed even though the 3741 attachment is busy or a 3741 not-ready/attachment check condition exists.

- The diagnostic byte is for CE diagnostics and has no meaning to the I/O control program.

- Figure 14-1 shows the 3741 status bits and their meanings.

- The directly attached 3741 should be powered up prior to initial program load (IPL) on System/3, and it should be powered off after the System/3 is powered off. Otherwise spurious (invalid) signals can be detected on the interface.

  These signals can momentarily set I/O transfer bits and/or the I/O ready status bit. In addition the interrupt pending status bit can also be set and, on the Model 15 only, an interrupt be generated.

Byte	Bit	Name	Indicates	Reset By
2	0	CE diagnostic	Used for CE diagnostic program.	CE action
2	1	CE diagnostic	Used for CE diagnostic program.	CE action
2	2	Interrupt pending	The 3741 requires program action.	Next SIO issued by program
2	3	Not used	Not used.	Not used
2	4	Data transfer register parity error	The 3741 attachment detected a parity error in at least 1 byte of data passing through the data transfer register for transmission to the read data field in main storage.	Next system reset, check reset, or SIO operation
2	5	No-op	The last instruction issued to the 3741 was rejected because the 3741 is not capable of performing the operation specified by the instruction.	Sense instruction
2	6	Length count register overflow	The operation tried to transfer more data than the number of bytes specified by the length count register.	Next LIO that reloads the length count register
2	7	I/O ready	The 3741 is ready.	3741 becoming not-ready
1	All	This byte contains the current contents of the length count register when the register was sensed.		

Figure 14-1. 3741 Status Bytes

INSTRUCTION FORMATS

Op	Mnemonic	Type
04	ZAZ	Two Address; Direct: Op, Q, Operand 1, Operand 2 (6 bytes)
06	AZ	
07	SZ	
08	MVX	
0A	ED	
0B	ITC	
0C	MVC	
0D	CLC	
0E	ALC	
0F	SLC	
14	ZAZ	Two Address; Direct, Indexed: Op, Q, Operand 1, D2 (5 bytes); XR1
16	AZ	
17	SZ	
18	MVX	
1A	ED	
1B	ITC	
1C	MVC	
1D	CLC	
1E	ALC	
1F	SLC	
24	ZAZ	Two Address; Direct, Indexed: Op, Q, Operand 1, D2 (5 bytes); XR2
26	AZ	
27	SZ	
28	MVX	
2A	ED	
2B	ITC	
2C	MVC	
2D	CLC	
2E	ALC	
2F	SLC	
30	SNS	One Address; Direct: Op, Q, Operand 1 (4 bytes)
31	LIO	
34	ST	
35	L	
36	A	
38	TBN	
39	TBF	
3A	SBN	
3B	SBF	
3C	MVI	
3D	CLI	
3E	SCP	
3F	LCP	
44	ZAZ	Two Address: Op, Q, D1, Operand 2 (5 bytes); XR1
46	AZ	
47	SZ	
48	MVX	
4A	ED	
4B	ITC	
4C	MVC	
4D	CLC	
4E	ALC	
4F	SLC	

Op	Mnemonic	Type
54	ZAZ	Two Address; Indexed: Op, Q, D1, D2 (4 bytes); XR1 XR1
56	AZ	
57	SZ	
58	MVX	
5A	ED	
5B	ITC	
5C	MVC	
5D	CLC	
5E	ALC	
5F	SLC	
64	ZAZ	Two Address: Op, Q, D1, D2 (4 bytes); XR1 XR2
66	AZ	
67	SZ	
68	MVX	
6A	ED	
6B	ITC	
6C	MVC	
6D	CLC	
6E	ALC	
6F	SLC	
70	SNS	One Address: Op, Q, D1 (3 bytes); XR1
71	LIO	
74	ST	
75	L	
76	A	
78	TBN	
79	TBF	
7A	SBN	
7B	SBF	
7C	MVI	
7D	CLI	
7E	SCP	
7F	LCP	
84	ZAZ	Two Address; Indexed, Direct: Op, Q, D1, Operand 2 (5 bytes); XR2
86	AZ	
87	SZ	
88	MVX	
8A	ED	
8B	ITC	
8C	MVC	
8D	CLC	
8E	ALC	
8F	SLC	

Op	Mnemonic	Type
94	ZAZ	Two Address; Indexed: Op, Q, D1, D2 (4 bytes); XR2 XR1
96	AZ	
97	SZ	
98	MVX	
9A	ED	
9B	ITC	
9C	MVC	
9D	CLC	
9E	ALC	
9F	SLC	
A4	ZAZ	Two Address; Indexed: Op, Q, D1, D2 (4 bytes); XR2 XR2
A6	AZ	
A7	SZ	
A8	MVX	
AA	ED	
AB	ITC	
AC	MVC	
AD	CLC	
AE	ALC	
AF	SLC	
B0	SNS	One Address; Indexed: Op, Q, D1 (3 bytes); XR2
B1	LIO	
B4	ST	
B5	L	
B6	A	
B8	TBN	
B9	TDF	
BA	SBN	
BB	SBF	
BC	MVI	
BD	CLI	
BE	SCP	
BF	LCP	
C0	BC	Direct: Op, Q, Address (4 bytes)
C1	TIO	
C2	LA	
D0	BC	Op, Q, D2 (3 bytes); +XR1
D1	TIO	
D2	LA	
E0	BC	Op, Q, D2 (3 bytes); +XR2
E1	TIO	
E2	LA	
F0	HPL	Op, Q, R (3 bytes)
F1	APL	
F2	JC	
F3	SIO	
F4	CCP	

INSTRUCTION FORMAT SUMMARY CHART 1

Bits 0-3	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	Q-Code (One Byte)	First Operand	Second Operand	Total Instr Length	Summary: Op	Summary: Q	Summary: Operand	
0					ZAZ		AZ	SZ	MVX		ED	ITC	MVC	CLC	ALC	SLC		2 Bytes direct	2 Bytes direct	6				
1					ZAZ		AZ	SZ	MVX		ED	ITC	MVC	CLC	ALC	SLC		2 Bytes direct	1 Byte displacement indexed by XR1	5				X1
2					ZAZ		AZ	SZ	MVX		ED	ITC	MVC	CLC	ALC	SLC		2 Bytes direct	1 Byte displacement indexed by XR2	5				X2
3	SNS	LIO			ST	L	A		TBN	TBF	SBN	SBF	MVI	CLI	SCP	LCP		2 Bytes direct		4				
4					ZAZ		AZ	SZ	MVX		ED	ITC	MVC	CLC	ALC	SLC		1 Byte displacement indexed by XR1	2 Bytes direct	5			X1	
5					ZAZ		AZ	SZ	MVX		ED	ITC	MVC	CLC	ALC	SLC		1 Byte displacement indexed by XR1	1 Byte displacement indexed by XR1	4			X1	X1
6					ZAZ		AZ	SZ	MVX		ED	ITC	MVC	CLC	ALC	SLC		1 Byte displacement indexed by XR1	1 Byte displacement indexed by XR2	4			X1	X2
7	SNS	LIO			ST	L	A		TBN	TBF	SBN	SBF	MVI	CLI	LCP	SCP		1 Byte displacement indexed by XR1		3			X1	
8					ZAZ		AZ	SZ	MVX		ED	ITC	MVC	CLC	ALC	SLC		1 Byte displacement indexed by XR2	2 Bytes direct	5			X2	
9					ZAZ		AZ	SZ	MVX		ED	ITC	MVC	CLC	ALC	SLC		1 Byte displacement indexed by XR2	1 Byte displacement indexed by XR1	4			X2	X1
A					ZAZ		AZ	SZ	MVX		ED	ITC	MVC	CLC	ALC	SLC		1 Byte displacement indexed by XR2	1 Byte displacement indexed by XR2	4			X2	X2
B	SNS	LIO			ST	L	A		TBN	TBF	SBN	SBF	MVI	CLI	SCP	LCP		1 Byte displacement indexed by XR2		3			X2	
C	BC	TIO	LA																2 Bytes direct	4				
D	BC	TIO	LA																1 Byte displacement indexed by XR1	3			X1	
E	BC	TIO	LA																1 Byte displacement indexed by XR2	3			X2	
F	HPL	APL	JC	SIO	CCP															3				

INSTRUCTION FORMAT SUMMARY CHART 2

INSTRUCTION TIMING

In the timing formulas,

N = Instruction length in bytes.

L1 = Length of destination field (two-address instruction) in bytes. Destination field is that field addressed by operand 1.

L2 = Length of source field (two-address instruction) in bytes. Source field is that field addressed by operand 2.

L = Length of the operand when the length of operand 1 must equal the length of operand 2.

R1 = Length of operand 1 when recomplementing is necessary.

M = Duration of single machine cycle.

Models 8, 10, 12, 15A, 15B, and 15C

Format	Instruction	Instruction Time	Length
Two-Address	Zero and add zoned Add zoned decimal Subtract zoned decimal	(N + L2 + L1)M + R1M	4, 5, or 6 bytes
	Move hex characters	(N + 2)M	
	Move characters Compare logical characters Add logical characters Subtract logical characters	(N + 2L)M	
	Insert and test characters	(N + 1 + L1)M, maximum	
	Edit	(N + L2 + L1)M	
One-Address	Move logical immediate Compare logical immediate Set bits on masked Test bits on masked Test bits off masked	(N + 1)M	3 or 4 bytes
	Store register Store CPU Load register Add to register Load CPU	(N + 2)M	
	Branch on condition Test I/O branch	NM	
	Load address	NM	
	Sense I/O Load I/O	(N + 2)M	
Command	Halt program level Advance program level—no-op Start I/O Jump on condition Command CPU	3M	3 bytes

Model 15D

Format	Instruction	Instruction Timings	
		4 or 6 bytes	3 or 5 bytes
Two-Address	Zero and add zoned Add zoned decimal Subtract zoned decimal	$\left[\frac{N}{2} + L2 + L1\right]$ M + R1 M	$\left[\frac{N+1}{2} + L2 + L1\right]$ M + R1 M
	Move hex characters	$\left[\frac{N}{2} + 2\right]$ M	$\left[\frac{N+1}{2} + 2\right]$ M
	Move characters Compare logical characters Add logical character Subtract logical character	$\left[\frac{N}{2} + 2L\right]$ M	$\left[\frac{N+1}{2} + 2L\right]$ M
	Edit	$\left[\frac{N}{2} + L1 + L2\right]$ M	$\left[\frac{N+1}{2} + L1 + L2\right]$ M
	Insert and test characters	$\left[\frac{N}{2} + 1 + L1\right]$ M, maximum	$\left[\frac{N+1}{2} + 1 + L1\right]$ M, maximum
One-Address	Move logical immediate Compare logical immediate Set bits on masked Test bits on masked Test bits off masked	$\left[\frac{N}{2} + 1\right]$ M	$\left[\frac{N+1}{2} + 1\right]$ M
	Store register Store CPU Load register Add to register Load CPU	$\left[\frac{N}{2} + 2\right]$ M	$\left[\frac{N+1}{2} + 2\right]$ M
	Branch on condition	$\frac{NM}{2}$	$\left[\frac{N+1}{2}\right]$ M
	Test I/O branch	NM	NM
	Load address	$\frac{NM}{2}$	$\left[\frac{N+1}{2}\right]$ M
	Sense I/O Load I/O	$[N + 2]$ M	$[N + 2]$ M
Command	Halt program level Advance program level Start I/O	3M	
	Jump on condition Command CPU	2M	

CODE CONVERSIONS

Dec Val	Hex Val	Card Code DCBA8421	Mnem	IPL (Note 1) T1T3	IPL (Note 1) T2T3	EBCDIC Code 01234567	EBCDIC Character	ASCII Code	ASCII Character	System/3 Symbol
000	00	C		4	1	00000000	NUL	0000000	NUL	
001	01	DCBA 1		A @	A 3	00000001	SOH	0000001	SOH	
002	02	DCBA 2		B @	B 3	00000010	STX	0000010	STX	
003	03	DCBA 21		C @	C 3	00000011	ETX	0000011	ETX	
004	04	DCBA 4	ZAZ	D @	D 3	00000100	PF	0000100	EOT	
005	05	DCBA 4 1		E @	E 3	00000101	HT	0000101	ENQ	
006	06	DCBA 42	AZ	F @	F 3	00000110	LC	0000110	ACK	
007	07	DCBA 421	SZ	G @	G 3	00000111	DEL	0000111	BEL	
008	08	DCBA8	MVX	H @	H 3	00001000		0001000	BS	
009	09	DCBA8 1		I @	I 3	00001001	RLF	0001001	HT	
010	0A	CBA8 2	ED	¢ 4	¢ 1	00001010	SMM	0001010	LF	
011	0B	CBA8 21	ITC	. 4	. 1	00001011	VT	0001011	VT	
012	0C	CBA84	MVC	< 4	< 1	00001100	FF	0001100	FF	
013	0D	CBA84 1	CLC	(4	(1	00001101	CR	0001101	CR	
014	0E	CBA842	ALC	+ 4	+ 1	00001110	SO	0001110	SO	
015	0F	CBA8421	SLC	| 4	| 1	00001111	SI	0001111	SI	
016	10	C A8 2		& 4	& 1	00010000	DLE	0010000	DLE	
017	11	DCB 1		J @	J 3	00010001	DC1	0010001	DC1	
018	12	DCB 2		K @	K 3	00010010	DC2	0010010	DC2	
019	13	DCB 21		L @	L 3	00010011	DC3(TM)	0010011	DC3	
020	14	DCB 4	ZAZ	M @	M 3	00010100	RES	0010100	DC4	
021	15	DCB 4 1		N @	N 3	00010101	NL	0010101	NAK	
022	16	DCB 42	AZ	O @	O 3	00010110	BS	0010110	SYN	
023	17	DCB 421	SZ	P @	P 3	00010111	IL	0010111	ETB	
024	18	DCB 8	MVX	Q @	Q 3	00011000	CAN	0011000	CAN	
025	19	DCB 8 1		R @	R 3	00011001	EM	0011001	EM	
026	1A	CB 8 2	ED	! 4	! 1	00011010	CC	0011010	SUB	
027	1B	CB 8 21	ITC	$ 4	$ 1	00011011	CUI	0011011	ESC	
028	1C	CB 84	MVC	* 4	* 1	00011100	IFS	0011100	FS	
029	1D	CB 84 1	CLC	) 4	) 1	00011101	IGS	0011101	GS	
030	1E	CB 842	ALC	; 4	; 1	00011110	IRS	0011110	RS	
031	1F	CB 8421	SLC	¬ 4	¬ 1	00011111	IUS*	0011111	US*	
032	20	CB		– 4	– 1	00100000	DS	0100000	SPACE	
033	21	C A 1		/ 4	/ 1	00100001	SOS	0100001	!	
034	22	DC A 2		S @	S 3	00100010	FS	0100010	"	
035	23	DC A 21		T @	T 3	00100011		0100011	#	
036	24	DC A 4	ZAZ	U @	U 3	00100100	BYP	0100100	$	
037	25	DC A 4 1		V @	V 3	00100101	LF	0100101	%	
038	26	DC A 42	AZ	W @	W 3	00100110	ETB(EOB)	0100110	&	
039	27	DC A 421	SZ	X @	X 3	00100111	ESC(PRE)	0100111	'	
040	28	DC A8	MVX	Y @	Y 3	00101000		0101000	(	
041	29	DC A8 1		Z @	Z 3	00101001		0101001	)	
042	2A	DCBA	ED	} @	} 3	00101010	SM	0101010	*	
043	2B	C A8 21	ITC	, 4	, 1	00101011	CU2	0101011	+	
044	2C	C A84	MVC	% 4	% 1	00101100		0101100	,	
045	2D	C A84 1	CLC	_ 4	_ 1	00101101	ENQ	0101101	-	
046	2E	C A842	ALC	> 4	> 1	00101110	ACK	0101110	.	
047	2F	C A8421	SLC	? 4	? 1	00101111	BEL	0101111	/	

* ITB character

Dec Val	Hex Val	Card Code DCBA8421	Mnem	IPL (Note 1) T1T3	IPL (Note 1) T2T3	EBCDIC Code 01234567	EBCDIC Character	ASCII Code 7654321	ASCII Character	System/3 Symbol
048	30	DC A	SNS	0 @	0 3	00110000		0110000	0	
049	31	DC 1	LIO	1 @	1 3	00110001		0110001	1	
050	32	DC 2		2 @	2 3	00110010	SYN	0110010	2	
051	33	DC 21		3 @	3 3	00110011		0110011	3	
052	34	DC 4	ST	4 @	4 3	00110100	PN	0110100	4	
053	35	DC 4 1	L	5 @	5 3	00110101	RS	0110101	5	
054	36	DC 42	A	6 @	6 3	00110110	UC	0110110	6	
055	37	DC 421		7 @	7 3	00110111	EOT	0110111	7	
056	38	DC 8	TBN	8 @	8 3	00111000		0111000	8	
057	39	DC 8 1	TBF	9 @	9 3	00111001		0111001	9	
058	3A	C 8 2	SBN	: 4	: 1	00111010		0111010	:	
059	3B	C 8 21	SBF	# 4	# 1	00111011	CU3	0111011	;	
060	3C	C 84	MVI	@ 4	@ 1	00111100	DC4	0111100	<	
061	3D	C 84 1	CLI	' 4	' 1	00111101	NAK	0111101	=	
062	3E	C 842	SCP	= 4	= 1	00111110		0111110	>	
063	3F	C 8421	LCP	" 4	" 1	00111111	SUB	0111111	?	
064	40	None				01000000	SPACE	1000000	@	SPACE
065	41	D BA 1		A 8	A 2	01000001		1000001	A	
066	42	D BA 2		B 8	B 2	01000010		1000010	B	
067	43	D BA 21		C 8	C 2	01000011		1000011	C	
068	44	D BA 4	ZAZ	D 8	D 2	01000100		1000100	D	
069	45	D BA 4 1		E 8	E 2	01000101		1000101	E	
070	46	D BA 42	AZ	F 8	F 2	01000110		1000110	F	
071	47	D BA 421	SZ	G 8	G 2	01000111		1000111	G	
072	48	D BA8	MVX	H 8	H 2	01001000		1001000	H	
073	49	D BA8 1		I 8	I 2	01001001		1001001	I	
074	4A	BA8 2	ED	¢	¢	01001010	¢	1001010	J	¢
075	4B	BA8 21	ITC	.	.	01001011	.	1001011	K	.
076	4C	BA84	MVC	<	<	01001100	<	1001100	L	<
077	4D	BA84 1	CLC	(	(	01001101	(	1001101	M	(
078	4E	BA842	ALC	+	+	01001110	+	1001110	N	+
079	4F	BA8421	SLC	|	|	01001111	|	1001111	O	|
080	50	A8 2		&	&	01010000	&	1010000	P	&
081	51	D B 1		J 8	J 2	01010001		1010001	Q	
082	52	D B 2		K 8	K 2	01010010		1010010	R	
083	53	D B 21		L 8	L 2	01010011		1010011	S	
084	54	D B 4	ZAZ	M 8	M 2	01010100		1010100	T	
085	55	D B 4 1		N 8	N 2	01010101		1010101	U	
086	56	D B 42	AZ	O 8	O 2	01010110		1010110	V	
087	57	D B 421	SZ	P 8	P 2	01010111		1010111	W	
088	58	D B 8	MVX	Q 8	Q 2	01011000		1011000	X	
089	59	D B 8 1		R 8	R 2	01011001		1011001	Y	
090	5A	B 8 2	ED	!	!	01011010	!	1011010	Z	!
091	5B	B 8 21	ITC	$	$	01011011	$	1011011	[	$
092	5C	B 84	MVC	*	*	01011100	*	1011100	\	*
093	5D	B 84 1	CLC	)	)	01011101	)	1011101	]	)
094	5E	B 842	ALC	;	;	01011110	;	1011110	¬	;
095	5F	B 8421	SLC	¬	¬	01011111	¬	1011111		¬

Dec Val	Hex Val	Card Code DCBA8421	Mnem	IPL (Note 1) T1T3	IPL (Note 1) T2T3	EBCDIC Code 01234567	EBCDIC Character	ASCII Code 7654321	ASCII Character	System/3 Symbol (Note 2)
096	60	B		–	–	01100000	–	1100000	`	–
097	61	A 1		/	/	01100001	/	1100001	a	/
098	62	D A 2		S 8	S 2	01100010		1100010	b	
099	63	D A 21		T 8	T 2	01100011		1100011	c	
100	64	D A 4	ZAZ	U 8	U 2	01100100		1100100	d	
101	65	D A 4 1		V 8	V 2	01100101		1100101	e	
102	66	D A 42	AZ	W 8	W 2	01100110		1100110	f	
103	67	D A 421	SZ	X 8	X 2	01100111		1100111	g	
104	68	D A8	MVX	Y 8	Y 2	01101000		1101000	h	
105	69	D A8 1		Z 8	Z 2	01101001		1101001	i	
106	6A	D BA	ED	} 8	} 2	01101010	¦	1101010	j	
107	6B	A8 21	ITC	,	,	01101011	,	1101011	k	,
108	6C	A84	MVC	%	%	01101100	%	1101100	l	%
109	6D	A84 1	CLC	_	_	01101101	_	1101101	m	_
110	6E	A842	ALC	>	>	01101110	>	1101110	n	>
111	6F	A8421	SLC	?	?	01101111	?	1101111	o	?
112	70	D A	SNS	0 8	0 2	01110000		1110000	p	
113	71	D 1	LIO	1 8	1 2	01110001		1110001	q	
114	72	D 2		2 8	2 2	01110010		1110010	r	
115	73	D 21		3 8	3 2	01110011		1110011	s	
116	74	D 4	ST	4 8	4 2	01110100		1110100	t	
117	75	D 4 1	L	5 8	5 2	01110101		1110101	u	
118	76	D 42	A	6 8	6 2	01110110		1110110	v	
119	77	D 421		7 8	7 2	01110111		1110111	w	
120	78	D 8	TBN	8 8	8 2	01111000		1111000	x	
121	79	D 8 1	TBF	9 8	9 2	01111001	`	1111001	y	
122	7A	8 2	SBN	:	:	01111010	:	1111010	z	:
123	7B	8 21	SBF	#	#	01111011	#	1111011	{	#
124	7C	84	MVI	@	@	01111100	@	1111100	¦	@
125	7D	84 1	CLI	'	'	01111101	'	1111101	}	'
126	7E	842	SCP	=	=	01111110	=	1111110	~	=
127	7F	8421	LCP	"	"	01111111	"	1111111	DEL	"
128	80	DC		@	3	10000000				
129	81	CBA 1		A 4	A 1	10000001	a			a
130	82	CBA 2		B 4	B 1	10000010	b			b
131	83	CBA 21		C 4	C 1	10000011	c			c
132	84	CBA 4	ZAZ	D 4	D 1	10000100	d			d
133	85	CBA 4 1		E 4	E 1	10000101	e			e
134	86	CBA 42	AZ	F 4	F 1	10000110	f			f
135	87	CBA 421	SZ	G 4	G 1	10000111	g			g
136	88	CBA8	MVX	H 4	H 1	10001000	h			h
137	89	CBA8 1		I 4	I 1	10001001	i			i
138	8A	DCBA8 2	ED	¢ @	¢ 3	10001010				
139	8B	DCBA8 21	ITC	. @	. 3	10001011				
140	8C	DCBA84	MVC	< @	< 3	10001100				Note 3 ≤
141	8D	DCBA84 1	CLC	(@	(3	10001101				(
142	8E	DCBA842	ALC	+ @	+ 3	10001110				+
143	8F	DCBA8421	SLC	| @	| 3	10001111				+

Dec Val	Hex Val	Card Code DCBA8421	Mnem	IPL (Note 1) T1T3	IPL (Note 1) T2T3	EBCDIC Code 01234567	EBCDIC Character	ASCII Code 7654321	ASCII Character	System/3 Symbol (Note 2)
144	90	CBA		} 4	} 1	10010000				
145	91	CB 1		J 4	J 1	10010001	j			j
146	92	CB 2		K 4	K 1	10010010	k			k
147	93	CB 21		L 4	L 1	10010011	l			l
148	94	CB 4	ZAZ	M 4	M 1	10010100	m			m
149	95	CB 4 1		N 4	N 1	10010101	n			n
150	96	CB 42	AZ	O 4	O 1	10010110	o			o
151	97	CB 421	SZ	P 4	P 1	10010111	p			p
152	98	CB 8	MVX	Q 4	Q 1	10011000	q			q
153	99	CB 8 1		R 4	R 1	10011001	r			r
154	9A	DCB 8 2	ED	! @	! 3	10011010				
155	9B	DCB 8 21	ITC	$ @	$ 3	10011011				
156	9C	DCB 84	MVC	* @	* 3	10011100				Note 3: ¤
157	9D	DCB 84 1	CLC	) @	) 3	10011101				Note 3:)
158	9E	DCB 842	ALC	; @	; 3	10011110				Note 3: ±
159	9F	DCB 8421	SLC	¬ @	¬ 3	10011111				Note 3: □
160	A0	DCB		– @	– 3	10100000				—
161	A1	DC A 1		/ @	/ 3	10100001	~			~
162	A2	C A 2		S 4	S 1	10100010	s			s
163	A3	C A 21		T 4	T 1	10100011	t			t
164	A4	C A 4	ZAZ	U 4	U 1	10100100	u			u
165	A5	C A 4 1		V 4	V 1	10100101	v			v
166	A6	C A 42	AZ	W 4	W 1	10100110	w			w
167	A7	C A 421	SZ	X 4	X 1	10100111	x			x
168	A8	C A8	MVX	Y 4	Y 1	10101000	y			y
169	A9	C A8 1		Z 4	Z 1	10101001	z			z
170	AA	DC A8 2	ED	& @	& 3	10101010				
171	AB	DC A8 21	ITC	, @	, 3	10101011				└
172	AC	DC A84	MVC	% @	% 3	10101100				┌
173	AD	DC A84 1	CLC	_ @	_ 3	10101101				[
174	AE	DC A842	ALC	> @	> 3	10101110				>
175	AF	DC A8421	SLC	? @	? 3	10101111				●
176	B0	C A	SNS	0 4	0 1	10110000				0
177	B1	C 1	LIO	1 4	1 1	10110001				1
178	B2	C 2		2 4	2 1	10110010				2
179	B3	C 21		3 4	3 1	10110011				3
180	B4	C 4	ST	4 4	4 1	10110100				4
181	B5	C 4 1	L	5 4	5 1	10110101				5
182	B6	C 42	A	6 4	6 1	10110110				6
183	B7	C 421		7 4	7 1	10110111				7
184	B8	C 8	TBN	8 4	8 1	10111000				8
185	B9	C 8 1	TBF	9 4	9 1	10111001				9
186	BA	DC 8 2	SBN	: @	: 3	10111010				
187	BB	DC 8 21	SBF	# @	# 3	10111011				┘
188	BC	DC 84	MVI	@ @	@ 3	10111100				┐
189	BD	DC 84 1	CLI	' @	' 3	10111101				]
190	BE	DC 842	SCP	= @	= 3	10111110				≠
191	BF	DC 8421	LCP	" @	" 3	10111111				—

Dec Val	Hex Val	Card Code DCBA8421	Mnem	IPL (Note 1) T1T3	IPL (Note 1) T2T3	EBCDIC Code 01234567	EBCDIC Character	ASCII Code 7654321	ASCII Character	System/3 Symbol (Note 2)
192	C0	D	BC	8	2	11000000	{			
193	C1	BA 1	TIO	A	A	11000001	A			A
194	C2	BA 2	LA	B	B	11000010	B			B
195	C3	BA 21		C	C	11000011	C			C
196	C4	BA 4		D	D	11000100	D			D
197	C5	BA 4 1		E	E	11000101	E			E
198	C6	BA 42		F	F	11000110	F			F
199	C7	BA 421		G	G	11000111	G			G
200	C8	BA8		H	H	11001000	H			H
201	C9	BA8 1		I	I	11001001	I			I
202	CA	D BA8 2		¢ 8	¢ 2	11001010				Note 3 } °
203	CB	D BA8 21		. 8	. 2	11001011				
204	CC	D BA84		< 8	< 2	11001100	[illegible] } Note 4			
205	CD	D BA84 1		(8	(2	11001101				
206	CE	D BA842		+ 8	+ 2	11001110	[illegible]			
207	CF	D BA8421		| 8	| 2	11001111				
208	D0	BA	BC	}	}	11010000	}			}
209	D1	B 1	TIO	J	J	11010001	J			J
210	D2	B 2	LA	K	K	11010010	K			K
211	D3	B 21		L	L	11010011	L			L
212	D4	B 4		M	M	11010100	M			M
213	D5	B 4 1		N	N	11010101	N			N
214	D6	B 42		O	O	11010110	O			O
215	D7	B 421		P	P	11010111	P			P
216	D8	B 8		Q	Q	11011000	Q			Q
217	D9	B 8 1		R	R	11011001	R			R
218	DA	D B 8 2		! 8	! 2	11011010				
219	DB	D B 8 21		$ 8	$ 2	11011011				
220	DC	D B 84		* 8	* 2	11011100				
221	DD	D B 84 1		) 8	) 2	11011101				
222	DE	D B 842		; 8	; 2	11011110				
223	DF	D B 8421		¬ 8	¬ 2	11011111				
224	E0	D B	BC	– 8	– 2	11100000	\			\
225	E1	D A 1	TIO	/ 8	/ 2	11100001				
226	E2	A 2	LA	S	S	11100010	S			S
227	E3	A 21		T	T	11100011	T			T
228	E4	A 4		U	U	11100100	U			U
229	E5	A 4 1		V	V	11100101	V			V
230	E6	A 42		W	W	11100110	W			W
231	E7	A 421		X	X	11100111	X			X
232	E8	A8		Y	Y	11101000	Y			Y
233	E9	A8 1		Z	Z	11101001	Z			Z
234	EA	D A8 2		& 8	& 2	11101010				
235	EB	D A8 21		, 8	, 2	11101011				

Dec Val	Hex Val	Card Code DCBA8421	Mnem	IPL (Note 1) T1T3	T2T3	EBCDIC Code 01234567	EBCDIC Character	ASCII Code 7654321	ASCII Character	System/3 Symbol (Note 2)
236	EC	D A84		% 8	% 2	11101100	⑂			
237	ED	D A84 1		— 8	— 2	11101101				
238	EE	D A842		> 8	> 2	11101110				
239	EF	D A8421		? 8	? 2	11101111				
240	F0	A	HPL	0	0	11110000	0			0
241	F1	1	APL	1	1	11110001	1			1
242	F2	2	JC	2	2	11110010	2			2
243	F3	21	SIO	3	3	11110011	3			3
244	F4	4	CCP	4	4	11110100	4			4
245	F5	4 1		5	5	11110101	5			5
246	F6	42		6	6	11110110	6			6
247	F7	421		7	7	11110111	7			7
248	F8	8		8	8	11111000	8			8
249	F9	8 1		9	9	11111001	9			9
250	FA	D 8 2		: 8	: 2	11111010	|			|
251	FB	D 8 21		# 8	# 2	11111011				
252	FC	D 84		@ 8	@ 2	11111100				
253	FD	D 84 1		' 8	' 2	11111101				
254	FE	D 842		= 8	= 2	11111110				
255	FF	D 8421		" 8	" 2	11111111				

Notes:

1. If both tier 1 and tier 2 are being used, the tier 3 punches are added together as shown in the following chart:

Tier 3 Character Required by Tier 1	Tier 3 Character Required by Tier 2
	1 2 3
4	5 6 7
8	9 : #
@	' = "

2. Characters on right side of column are not handled by 6-bit devices.
3. Symbols printed by System/3 devices equipped with Tn character sets. 8D, 8E, 9D, A0, and B0 through B9 are superscript characters.
4. Special graphics.

POWERS OF 2 TABLE

2^n	n	2^{-n}
1	0	1.0
2	1	0.5
4	2	0.25
8	3	0.125
16	4	0.062 5
32	5	0.031 25
64	6	0.015 625
128	7	0.007 812 5
256	8	0.003 906 25
512	9	0.001 953 125
1 024	10	0.000 976 562 5
2 048	11	0.000 488 281 25
4 096	12	0.000 244 140 625
8 192	13	0.000 122 070 312 5
16 384	14	0.000 061 035 156 25
32 768	15	0.000 030 517 578 125
65 536	16	0.000 015 258 789 062 5
131 072	17	0.000 007 629 394 531 25
262 144	18	0.000 003 814 697 265 625
524 288	19	0.000 001 907 348 632 812 5
1 048 576	20	0.000 000 953 674 316 406 25
2 097 152	21	0.000 000 476 837 158 203 125
4 194 304	22	0.000 000 238 418 579 101 562 5
8 388 608	23	0.000 000 119 209 289 550 781 25
16 777 216	24	0.000 000 059 604 644 775 390 625
33 554 432	25	0.000 000 029 802 322 387 695 312 5
67 108 864	26	0.000 000 014 901 161 193 847 656 25
134 217 728	27	0.000 000 007 450 580 596 923 828 125
268 435 456	28	0.000 000 003 725 290 298 461 914 062 5
536 870 912	29	0.000 000 001 862 645 149 230 957 031 25
1 073 741 824	30	0.000 000 000 931 322 574 615 478 515 625
2 147 483 648	31	0.000 000 000 465 661 287 307 739 257 812 5
4 294 967 296	32	0.000 000 000 232 830 643 653 869 628 906 25
8 589 934 592	33	0.000 000 000 116 415 321 826 934 814 453 125
17 179 869 184	34	0.000 000 000 058 207 660 913 467 407 226 562 5
34 359 738 368	35	0.000 000 000 029 103 830 456 733 703 613 281 25
68 719 476 736	36	0.000 000 000 014 551 915 228 366 851 806 640 625
137 438 953 472	37	0.000 000 000 007 275 957 614 183 425 903 320 312 5
274 877 906 944	38	0.000 000 000 003 637 978 807 091 712 951 660 156 25
549 755 813 888	39	0.000 000 000 001 818 989 403 545 856 475 830 078 125

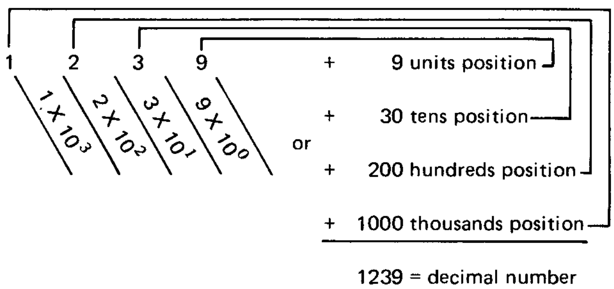
BINARY AND HEXADECIMAL NUMBER NOTATION

Binary Number Notation

A binary number system, such as is used in System/3, uses a base of 2. The concept of using a base of 2 can be compared with the base of 10 (decimal) number system.

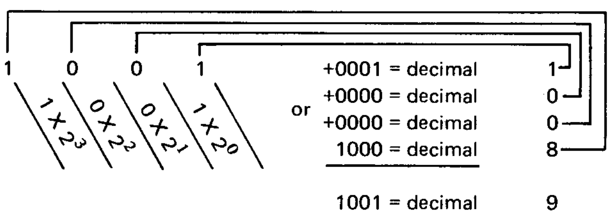
Decimal Number	Binary Number
0	0
1	1
2	10
3	11
4	100
5	101
6	110
7	111
8	1000
9	1001

Example of a binary number:



As shown above, the decimal number system allows counting to 10 in each position—from units to tens to hundreds to thousands, etc. The binary system allows counting to 2 in each position. Register displays in the System/3 are in binary forms: a bit light on is a 1; a bit light off is a 0.

Example of a decimal number:



Hexadecimal Number System

It has been noted that binary numbers require about three times as many positions as decimal numbers to express the equivalent number. This is not much of a problem to the computer; however, in talking and writing or in communicating with the computer, these binary numbers are bulky.

A long string of 1's and 0's cannot be effectively transmitted from one individual to another. Some shorthand method is necessary.

The hexadecimal number system fills this need. Because of the simple relationship of hexadecimal to binary, numbers can be converted from one system to another by inspection. The base or radix of the hexadecimal system is 16. This means there are 16 symbols: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, and F. The letters A, B, C, D, E, and F represent the 10-base system values of 10, 11, 12, 13, 14, and 15, respectively.

Four binary positions are equivalent to one hexadecimal position. The following table shows the comparable values of the three number systems.

Decimal	Binary	Hexadecimal
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F

At this point all 16 symbols have been used, and a carry to the next higher position of the number is necessary. For example:

Decimal	Binary	Hexadecimal
16	0001 0000	10
17	0001 0001	11
18	0001 0010	12
19	0001 0011	13
20	0001 0100	14
21	0001 0101	15

– and so on –

Remember that as far as the internal circuitry of the computer is concerned, it understands only binary. But an operator can look at a series of lights on the computer console showing binary 1's and 0's, for example: 0001 1110 0001 0011, and say that the lights represent the hexadecimal value 1E13, which is easier to state than the string of 1's and 0's.

HEXADECIMAL–DECIMAL CONVERSION TABLES

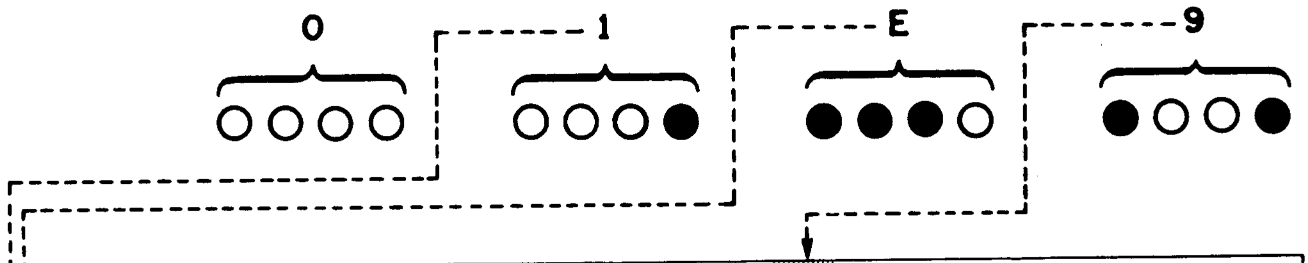
The table in this appendix provides for direct conversion of decimal and hexadecimal number in these ranges:

Hexadecimal	Decimal
000 to FFF	0000 to 4095

For numbers outside the range of the table, add the following values to the table figures:

Hexadecimal	Decimal
1000	4096
2000	8192
3000	12288

Hexadecimal	Decimal
4000	16384
5000	20480
6000	24576
7000	28672
8000	32768



	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00_	0000	0001	0002	0003	0004	0005	0006	0007	0008	0009	0010	0011	0012	0013	0014	0015
01_	0016	0017	0018	0019	0020	0021	0022	0023	0024	0025	0026	0027	0028	0029	0030	0031
02_	0032	0033	0034	0035	0036	0037	0038	0039	0040	0041	0042	0043	0044	0045	0046	0047
03_	0048	0049	0050	0051	0052	0053	0054	0055	0056	0057	0058	0059	0060	0061	0062	0063
04_	0064	0065	0066	0067	0068	0069	0070	0071	0072	0073	0074	0075	0076	0077	0078	0079
05_	0080	0081	0082	0083	0084	0085	0086	0087	0088	0089	0090	0091	0092	0093	0094	0095
06_	0096	0097	0098	0099	0100	0101	0102	0103	0104	0105	0106	0107	0108	0109	0110	0111
07_	0112	0113	0114	0115	0116	0117	0118	0119	0120	0121	0122	0123	0124	0125	0126	0127
08_	0128	0129	0130	0131	0132	0133	0134	0135	0136	0137	0138	0139	0140	0141	0142	0143
09_	0144	0145	0146	0147	0148	0149	0150	0151	0152	0153	0154	0155	0156	0157	0158	0159
0A_	0160	0161	0162	0163	0164	0165	0166	0167	0168	0169	0170	0171	0172	0173	0174	0175
0B_	0176	0177	0178	0179	0180	0181	0182	0183	0184	0185	0186	0187	0188	0189	0190	0191
0C_	0192	0193	0194	0195	0196	0197	0198	0199	0200	0201	0202	0203	0204	0205	0206	0207
0D_	0208	0209	0210	0211	0212	0213	0214	0215	0216	0217	0218	0219	0220	0221	0222	0223
0E_	0224	0225	0226	0227	0228	0229	0230	0231	0232	0233	0234	0235	0236	0237	0238	0239
0F_	0240	0241	0242	0243	0244	0245	0246	0247	0248	0249	0250	0251	0252	0253	0254	0255
10_	0256	0257	0258	0259	0260	0261	0262	0263	0264	0265	0266	0267	0268	0269	0270	0271
11_	0272	0273	0274	0275	0276	0277	0278	0279	0280	0281	0282	0283	0284	0285	0286	0287
12_	0288	0289	0290	0291	0292	0293	0294	0295	0296	0297	0298	0299	0300	0301	0302	0303
13_	0304	0305	0306	0307	0308	0309	0310	0311	0312	0313	0314	0315	0316	0317	0318	0319
14_	0320	0321	0322	0323	0324	0325	0326	0327	0328	0329	0330	0331	0332	0333	0334	0335
15_	0336	0337	0338	0339	0340	0341	0342	0343	0344	0345	0346	0347	0348	0349	0350	0351
16_	0352	0353	0354	0355	0356	0357	0358	0359	0360	0361	0362	0363	0364	0365	0366	0367
17_	0368	0369	0370	0371	0372	0373	0374	0375	0376	0377	0378	0379	0380	0381	0382	0383
18_	0384	0385	0386	0387	0388	0389	0390	0391	0392	0393	0394	0395	0396	0397	0398	0399
19_	0400	0401	0402	0403	0404	0405	0406	0407	0408	0409	0410	0411	0412	0413	0414	0415
1A_	0416	0417	0418	0419	0420	0421	0422	0423	0424	0425	0426	0427	0428	0429	0430	0431
1B_	0432	0433	0434	0435	0436	0437	0438	0439	0440	0441	0442	0443	0444	0445	0446	0447
1C_	0448	0449	0450	0451	0452	0453	0454	0455	0456	0457	0458	0459	0460	0461	0462	0463
1D_	0464	0465	0466	0467	0468	0469	0470	0471	0472	0473	0474	0475	0476	0477	0478	0479
1E_	0480	0481	0482	0483	0484	0485	0486	0487	0488	0489	0490	0491	0492	0493	0494	0495
1F_	0496	0497	0498	0499	0500	0501	0502	0503	0504	0505	0506	0507	0508	0509	0510	0511

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
20_	0512	0513	0514	0515	0516	0517	0518	0519	0520	0521	0522	0523	0524	0525	0526	0527
21_	0528	0529	0530	0531	0532	0533	0534	0535	0536	0537	0538	0539	0540	0541	0542	0543
22_	0544	0545	0546	0547	0548	0549	0550	0551	0552	0553	0554	0555	0556	0557	0558	0559
23_	0560	0561	0562	0563	0564	0565	0566	0567	0568	0569	0570	0571	0572	0573	0574	0575
24_	0576	0577	0578	0579	0580	0581	0582	0583	0584	0585	0586	0587	0588	0589	0590	0591
25_	0592	0593	0594	0595	0596	0597	0598	0599	0600	0601	0602	0603	0604	0605	0606	0607
26_	0608	0609	0610	0611	0612	0613	0614	0615	0616	0617	0618	0619	0620	0621	0622	0623
27_	0624	0625	0626	0627	0628	0629	0630	0631	0632	0633	0634	0635	0636	0637	0638	0639
28_	0640	0641	0642	0643	0644	0645	0646	0647	0648	0649	0650	0651	0652	0653	0654	0655
29_	0656	0657	0658	0659	0660	0661	0662	0663	0664	0665	0666	0667	0668	0669	0670	0671
2A_	0672	0673	0674	0675	0676	0677	0678	0679	0680	0681	0682	0683	0684	0685	0686	0687
2B_	0688	0689	0690	0691	0692	0693	0694	0695	0696	0697	0698	0699	0700	0701	0702	0703
2C_	0704	0705	0706	0707	0708	0709	0710	0711	0712	0713	0714	0715	0716	0717	0718	0719
2D_	0720	0721	0722	0723	0724	0725	0726	0727	0728	0729	0730	0731	0732	0733	0734	0735
2E_	0736	0737	0738	0739	0740	0741	0742	0743	0744	0745	0746	0747	0748	0749	0750	0751
2F_	0752	0753	0754	0755	0756	0757	0758	0759	0760	0761	0762	0763	0764	0765	0766	0767
30_	0768	0769	0770	0771	0772	0773	0774	0775	0776	0777	0778	0779	0780	0781	0782	0783
31_	0784	0785	0786	0787	0788	0789	0790	0791	0792	0793	0794	0795	0796	0797	0798	0799
32_	0800	0801	0802	0803	0804	0805	0806	0807	0808	0809	0810	0811	0812	0813	0814	0815
33_	0816	0817	0818	0819	0820	0821	0822	0823	0824	0825	0826	0827	0828	0829	0830	0831
34_	0832	0833	0834	0835	0836	0837	0838	0839	0840	0841	0842	0843	0844	0845	0846	0847
35_	0848	0849	0850	0851	0852	0853	0854	0855	0856	0857	0858	0859	0860	0861	0862	0863
36_	0864	0865	0866	0867	0868	0869	0870	0871	0872	0873	0874	0875	0876	0877	0878	0879
37_	0880	0881	0882	0883	0884	0885	0886	0887	0888	0889	0890	0891	0892	0893	0894	0895
38_	0896	0897	0898	0899	0900	0901	0902	0903	0904	0905	0906	0907	0908	0909	0910	0911
39_	0912	0913	0914	0915	0916	0917	0918	0919	0920	0921	0922	0923	0924	0925	0926	0927
3A_	0928	0929	0930	0931	0932	0933	0934	0935	0936	0937	0938	0939	0940	0941	0942	0943
3B_	0944	0945	0946	0947	0948	0949	0950	0951	0952	0953	0954	0955	0956	0957	0958	0959
3C_	0960	0961	0962	0963	0964	0965	0966	0967	0968	0969	0970	0971	0972	0973	0974	0975
3D_	0976	0977	0978	0979	0980	0981	0982	0983	0984	0985	0986	0987	0988	0989	0990	0991
3E_	0992	0993	0994	0995	0996	0997	0998	0999	1000	1001	1002	1003	1004	1005	1006	1007
3F_	1008	1009	1010	1011	1012	1013	1014	1015	1016	1017	1018	1019	1020	1021	1022	1023

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
40_	1024	1025	1026	1027	1028	1029	1030	1031	1032	1033	1034	1035	1036	1037	1038	1039
41_	1040	1041	1042	1043	1044	1045	1046	1047	1048	1049	1050	1051	1052	1053	1054	1055
42_	1056	1057	1058	1059	1060	1061	1062	1063	1064	1065	1066	1067	1068	1069	1070	1071
43_	1072	1073	1074	1075	1076	1077	1078	1079	1080	1081	1082	1083	1084	1085	1086	1087
44_	1088	1089	1090	1091	1092	1093	1094	1095	1096	1097	1098	1099	1100	1101	1102	1103
45_	1104	1105	1106	1107	1108	1109	1110	1111	1112	1113	1114	1115	1116	1117	1118	1119
46_	1120	1121	1122	1123	1124	1125	1126	1127	1128	1129	1130	1131	1132	1133	1134	1135
47_	1136	1137	1138	1139	1140	1141	1142	1143	1144	1145	1146	1147	1148	1149	1150	1151
48_	1152	1153	1154	1155	1156	1157	1158	1159	1160	1161	1162	1163	1164	1165	1166	1167
49_	1168	1169	1170	1171	1172	1173	1174	1175	1176	1177	1178	1179	1180	1181	1182	1183
4A_	1184	1185	1186	1187	1188	1189	1190	1191	1192	1193	1194	1195	1196	1197	1198	1199
4B_	1200	1201	1202	1203	1204	1205	1206	1207	1208	1209	1210	1211	1212	1213	1214	1215
4C_	1216	1217	1218	1219	1220	1221	1222	1223	1224	1225	1226	1227	1228	1229	1230	1231
4D_	1232	1233	1234	1235	1236	1237	1238	1239	1240	1241	1242	1243	1244	1245	1246	1247
4E_	1248	1249	1250	1251	1252	1253	1254	1255	1256	1257	1258	1259	1260	1261	1262	1263
4F_	1264	1265	1266	1267	1268	1269	1270	1271	1272	1273	1274	1275	1276	1277	1278	1279
50_	1280	1281	1282	1283	1284	1285	1286	1287	1288	1289	1290	1291	1292	1293	1294	1295
51_	1296	1297	1298	1299	1300	1301	1302	1303	1304	1305	1306	1307	1308	1309	1310	1311
52_	1312	1313	1314	1315	1316	1317	1318	1319	1320	1321	1322	1323	1324	1325	1326	1327
53_	1328	1329	1330	1331	1332	1333	1334	1335	1336	1337	1338	1339	1340	1341	1342	1343
54_	1344	1345	1346	1347	1348	1349	1350	1351	1352	1353	1354	1355	1356	1357	1358	1359
55_	1360	1361	1362	1363	1364	1365	1366	1367	1368	1369	1370	1371	1372	1373	1374	1375
56_	1376	1377	1378	1379	1380	1381	1382	1383	1384	1385	1386	1387	1388	1389	1390	1391
57_	1392	1393	1394	1395	1396	1397	1398	1399	1400	1401	1402	1403	1404	1405	1406	1407
58_	1408	1409	1410	1411	1412	1413	1414	1415	1416	1417	1418	1419	1420	1421	1422	1423
59_	1424	1425	1426	1427	1428	1429	1430	1431	1432	1433	1434	1435	1436	1437	1438	1439
5A_	1440	1441	1442	1443	1444	1445	1446	1447	1448	1449	1450	1451	1452	1453	1454	1455
5B_	1456	1457	1458	1459	1460	1461	1462	1463	1464	1465	1466	1467	1468	1469	1470	1471
5C_	1472	1473	1474	1475	1476	1477	1478	1479	1480	1481	1482	1483	1484	1485	1486	1487
5D_	1488	1489	1490	1491	1492	1493	1494	1495	1496	1497	1498	1499	1500	1501	1502	1503
5E_	1504	1505	1506	1507	1508	1509	1510	1511	1512	1513	1514	1515	1516	1517	1518	1519
5F_	1520	1521	1522	1523	1524	1525	1526	1527	1528	1529	1530	1531	1532	1533	1534	1535

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
60_	1536	1537	1538	1539	1540	1541	1542	1543	1544	1545	1546	1547	1548	1549	1550	1551
61_	1552	1553	1554	1555	1556	1557	1558	1559	1560	1561	1562	1563	1564	1565	1566	1567
62_	1568	1569	1570	1571	1572	1573	1574	1575	1576	1577	1578	1579	1580	1581	1582	1583
63_	1584	1585	1586	1587	1588	1589	1590	1591	1592	1593	1594	1595	1596	1597	1598	1599
64_	1600	1601	1602	1603	1604	1605	1606	1607	1608	1609	1610	1611	1612	1613	1614	1615
65_	1616	1617	1618	1619	1620	1621	1622	1623	1624	1625	1626	1627	1628	1629	1630	1631
66_	1632	1633	1634	1635	1636	1637	1638	1639	1640	1641	1642	1643	1644	1645	1646	1647
67_	1648	1649	1650	1651	1652	1653	1654	1655	1656	1657	1658	1659	1660	1661	1662	1663
68_	1664	1665	1666	1667	1668	1669	1670	1671	1672	1673	1674	1675	1676	1677	1678	1679
69_	1680	1681	1682	1683	1684	1685	1686	1687	1688	1689	1690	1691	1692	1693	1694	1695
6A_	1696	1697	1698	1699	1700	1701	1702	1703	1704	1705	1706	1707	1708	1709	1710	1711
6B_	1712	1713	1714	1715	1716	1717	1718	1719	1720	1721	1722	1723	1724	1725	1726	1727
6C_	1728	1729	1730	1731	1732	1733	1734	1735	1736	1737	1738	1739	1740	1741	1742	1743
6D_	1744	1745	1746	1747	1748	1749	1750	1751	1752	1753	1754	1755	1756	1757	1758	1759
6E_	1760	1761	1762	1763	1764	1765	1766	1767	1768	1769	1770	1771	1772	1773	1774	1775
6F_	1776	1777	1778	1779	1780	1781	1782	1783	1784	1785	1786	1787	1788	1789	1790	1791
70_	1792	1793	1794	1795	1796	1797	1798	1799	1800	1801	1802	1803	1804	1805	1806	1807
71_	1808	1809	1810	1811	1812	1813	1814	1815	1816	1817	1818	1819	1820	1821	1822	1823
72_	1824	1825	1826	1827	1828	1829	1830	1831	1832	1833	1834	1835	1836	1837	1838	1839
73_	1840	1841	1842	1843	1844	1845	1846	1847	1848	1849	1850	1851	1852	1853	1854	1855
74_	1856	1857	1858	1859	1860	1861	1862	1863	1864	1865	1866	1867	1868	1869	1870	1871
75_	1872	1873	1874	1875	1876	1877	1878	1879	1880	1881	1882	1883	1884	1885	1886	1887
76_	1888	1889	1890	1891	1892	1893	1894	1895	1896	1897	1898	1899	1900	1901	1902	1903
77_	1904	1905	1906	1907	1908	1909	1910	1911	1912	1913	1914	1915	1916	1917	1918	1919
78_	1920	1921	1922	1923	1924	1925	1926	1927	1928	1929	1930	1931	1932	1933	1934	1935
79_	1936	1937	1938	1939	1940	1941	1942	1943	1944	1945	1946	1947	1948	1949	1950	1951
7A_	1952	1953	1954	1955	1956	1957	1958	1959	1960	1961	1962	1963	1964	1965	1966	1967
7B_	1968	1969	1970	1971	1972	1973	1974	1975	1976	1977	1978	1979	1980	1981	1982	1983
7C_	1984	1985	1986	1987	1988	1989	1990	1991	1992	1993	1994	1995	1996	1997	1998	1999
7D_	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013	2014	2015
7E_	2016	2017	2018	2019	2020	2021	2022	2023	2024	2025	2026	2027	2028	2029	2030	2031
7F_	2032	2033	2034	2035	2036	2037	2038	2039	2040	2041	2042	2043	2044	2045	2046	2047

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
80_	2048	2049	2050	2051	2052	2053	2054	2055	2056	2057	2058	2059	2060	2061	2062	2063
81_	2064	2065	2066	2067	2068	2069	2070	2071	2072	2073	2074	2075	2076	2077	2078	2079
82_	2080	2081	2082	2083	2084	2085	2086	2087	2088	2089	2090	2091	2092	2093	2094	2095
83_	2096	2097	2098	2099	2100	2101	2102	2103	2104	2105	2106	2107	2108	2109	2110	2111
84_	2112	2113	2114	2115	2116	2117	2118	2119	2120	2121	2122	2123	2124	2125	2126	2127
85_	2128	2129	2130	2131	2132	2133	2134	2135	2136	2137	2138	2139	2140	2141	2142	2143
86_	2144	2145	2146	2147	2148	2149	2150	2151	2152	2153	2154	2155	2156	2157	2158	2159
87_	2160	2161	2162	2163	2164	2165	2166	2167	2168	2169	2170	2171	2172	2173	2174	2175
88_	2176	2177	2178	2179	2180	2181	2182	2183	2184	2185	2186	2187	2188	2189	2190	2191
89_	2192	2193	2194	2195	2196	2197	2198	2199	2200	2201	2202	2203	2204	2205	2206	2207
8A_	2208	2209	2210	2211	2212	2213	2214	2215	2216	2217	2218	2219	2220	2221	2222	2223
8B_	2224	2225	2226	2227	2228	2229	2230	2231	2232	2233	2234	2235	2236	2237	2238	2239
8C_	2240	2241	2242	2243	2244	2245	2246	2247	2248	2249	2250	2251	2252	2253	2254	2255
8D_	2256	2257	2258	2259	2260	2261	2262	2263	2264	2265	2266	2267	2268	2269	2270	2271
8E_	2272	2273	2274	2275	2276	2277	2278	2279	2280	2281	2282	2283	2284	2285	2286	2287
8F_	2288	2289	2290	2291	2292	2293	2294	2295	2296	2297	2298	2299	2300	2301	2302	2303
90_	2304	2305	2306	2307	2308	2309	2310	2311	2312	2313	2314	2315	2316	2317	2318	2319
91_	2320	2321	2322	2323	2324	2325	2326	2327	2328	2329	2330	2331	2332	2333	2334	2335
92_	2336	2337	2338	2339	2340	2341	2342	2343	2344	2345	2346	2347	2348	2349	2350	2351
93_	2352	2353	2354	2355	2356	2357	2358	2359	2360	2361	2362	2363	2364	2365	2366	2367
94_	2368	2369	2370	2371	2372	2373	2374	2375	2376	2377	2378	2379	2380	2381	2382	2383
95_	2384	2385	2386	2387	2388	2389	2390	2391	2392	2393	2394	2395	2396	2397	2398	2399
96_	2400	2401	2402	2403	2404	2405	2406	2407	2408	2409	2410	2411	2412	2413	2414	2415
97_	2416	2417	2418	2419	2420	2421	2422	2423	2424	2425	2426	2427	2428	2429	2430	2431
98_	2432	2433	2434	2435	2436	2437	2438	2439	2440	2441	2442	2443	2444	2445	2446	2447
99_	2448	2449	2450	2451	2452	2453	2454	2455	2456	2457	2458	2459	2460	2461	2462	2463
9A_	2464	2465	2466	2467	2468	2469	2470	2471	2472	2473	2474	2475	2476	2477	2478	2479
9B_	2480	2481	2482	2483	2484	2485	2486	2487	2488	2489	2490	2491	2492	2493	2494	2495
9C_	2496	2497	2498	2499	2500	2501	2502	2503	2504	2505	2506	2507	2508	2509	2510	2511
9D_	2512	2513	2514	2515	2516	2517	2518	2519	2520	2521	2522	2523	2524	2525	2526	2527
9E_	2528	2529	2530	2531	2532	2533	2534	2535	2536	2537	2538	2539	2540	2541	2542	2543
9F_	2544	2545	2546	2547	2548	2549	2550	2551	2552	2553	2554	2555	2556	2557	2558	2559

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
A0_	2560	2561	2562	2563	2564	2565	2566	2567	2568	2569	2570	2571	2572	2573	2574	2575
A1_	2576	2577	2578	2579	2580	2581	2582	2583	2584	2585	2586	2587	2588	2589	2590	2591
A2_	2592	2593	2594	2595	2596	2597	2598	2599	2600	2601	2602	2603	2604	2605	2606	2607
A3_	2608	2609	2610	2611	2612	2613	2614	2615	2616	2617	2618	2619	2620	2621	2622	2623
A4_	2624	2625	2626	2627	2628	2629	2630	2631	2632	2633	2634	2635	2636	2637	2638	2639
A5_	2640	2641	2642	2643	2644	2645	2646	2647	2648	2649	2650	2651	2652	2653	2654	2655
A6_	2656	2657	2658	2659	2660	2661	2662	2663	2664	2665	2666	2667	2668	2669	2670	2671
A7_	2672	2673	2674	2675	2676	2677	2678	2679	2680	2681	2682	2683	2684	2685	2686	2687
A8_	2688	2689	2690	2691	2692	2693	2694	2695	2696	2697	2698	2699	2700	2701	2702	2703
A9_	2704	2705	2706	2707	2708	2709	2710	2711	2712	2713	2714	2715	2716	2717	2718	2719
AA_	2720	2721	2722	2723	2724	2725	2726	2727	2728	2729	2730	2731	2732	2733	2734	2735
AB_	2736	2737	2738	2739	2740	2741	2742	2743	2744	2745	2746	2747	2748	2749	2750	2751
AC_	2752	2753	2754	2755	2756	2757	2758	2759	2760	2761	2762	2763	2764	2765	2766	2767
AD_	2768	2769	2770	2771	2772	2773	2774	2775	2776	2777	2778	2779	2780	2781	2782	2783
AE_	2784	2785	2786	2787	2788	2789	2790	2791	2792	2793	2794	2795	2796	2797	2798	2799
AF_	2800	2801	2802	2803	2804	2805	2806	2807	2808	2809	2810	2811	2812	2813	2814	2815
B0_	2816	2817	2818	2819	2820	2821	2822	2823	2824	2825	2826	2827	2828	2829	2830	2831
B1_	2832	2833	2834	2835	2836	2837	2838	2839	2840	2841	2842	2843	2844	2845	2846	2847
B2_	2848	2849	2850	2851	2852	2853	2854	2855	2856	2857	2858	2859	2860	2861	2862	2863
B3_	2864	2865	2866	2867	2868	2869	2870	2871	2872	2873	2874	2875	2876	2877	2878	2879
B4_	2880	2881	2882	2883	2884	2885	2886	2887	2888	2889	2890	2891	2892	2893	2894	2895
B5_	2896	2897	2898	2899	2900	2901	2902	2903	2904	2905	2906	2907	2908	2909	2910	2911
B6_	2912	2913	2914	2915	2916	2917	2918	2919	2920	2921	2922	2923	2924	2925	2926	2927
B7_	2928	2929	2930	2931	2932	2933	2934	2935	2936	2937	2938	2939	2940	2941	2942	2943
B8_	2944	2945	2946	2947	2948	2949	2950	2951	2952	2953	2954	2955	2956	2957	2958	2959
B9_	2960	2961	2962	2963	2964	2965	2966	2967	2968	2969	2970	2971	2972	2973	2974	2975
BA_	2976	2977	2978	2979	2980	2981	2982	2983	2984	2985	2986	2987	2988	2989	2990	2991
BB_	2992	2993	2994	2995	2996	2997	2998	2999	3000	3001	3002	3003	3004	3005	3006	3007
BC_	3008	3009	3010	3011	3012	3013	3014	3015	3016	3017	3018	3019	3020	3021	3022	3023
BD_	3024	3025	3026	3027	3028	3029	3030	3031	3032	3033	3034	3035	3036	3037	3038	3039
BE_	3040	3041	3042	3043	3044	3045	3046	3047	3048	3049	3050	3051	3052	3053	3054	3055
BF_	3056	3057	3058	3059	3060	3061	3062	3063	3064	3065	3066	3067	3068	3069	3070	3071

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
C0_	3072	3073	3074	3075	3076	3077	3078	3079	3080	3081	3082	3083	3084	3085	3086	3087
C1_	3088	3089	3090	3091	3092	3093	3094	3095	3096	3097	3098	3099	3100	3101	3102	3103
C2_	3104	3105	3106	3107	3108	3109	3110	3111	3112	3113	3114	3115	3116	3117	3118	3119
C3_	3120	3121	3122	3123	3124	3125	3126	3127	3128	3129	3130	3131	3132	3133	3134	3135
C4_	3136	3137	3138	3139	3140	3141	3142	3143	3144	3145	3146	3147	3148	3149	3150	3151
C5_	3152	3153	3154	3155	3156	3157	3158	3159	3160	3161	3162	3163	3164	3165	3166	3167
C6_	3168	3169	3170	3171	3172	3173	3174	3175	3176	3177	3178	3179	3180	3181	3182	3183
C7_	3184	3185	3186	3187	3188	3189	3190	3191	3192	3193	3194	3195	3196	3197	3198	3199
C8_	3200	3201	3202	3203	3204	3205	3206	3207	3208	3209	3210	3211	3212	3213	3214	3215
C9_	3216	3217	3218	3219	3220	3221	3222	3223	3224	3225	3226	3227	3228	3229	3230	3231
CA_	3232	3233	3234	3235	3236	3237	3238	3239	3240	3241	3242	3243	3244	3245	3246	3247
CB_	3248	3249	3250	3251	3252	3253	3254	3255	3256	3257	3258	3259	3260	3261	3262	3263
CC_	3264	3265	3266	3267	3268	3269	3270	3271	3272	3273	3274	3275	3276	3277	3278	3279
CD_	3280	3281	3282	3283	3284	3285	3286	3287	3288	3289	3290	3291	3292	3293	3294	3295
CE_	3296	3297	3298	3299	3300	3301	3302	3303	3304	3305	3306	3307	3308	3309	3310	3311
CF_	3312	3313	3314	3315	3316	3317	3318	3319	3320	3321	3322	3323	3324	3325	3326	3327
D0_	3328	3329	3330	3331	3332	3333	3334	3335	3336	3337	3338	3339	3340	3341	3342	3343
D1_	3344	3345	3346	3347	3348	3349	3350	3351	3352	3353	3354	3355	3356	3357	3358	3359
D2_	3360	3361	3362	3363	3364	3365	3366	3367	3368	3369	3370	3371	3372	3373	3374	3375
D3_	3376	3377	3378	3379	3380	3381	3382	3383	3384	3385	3386	3387	3388	3389	3390	3391
D4_	3392	3393	3394	3395	3396	3397	3398	3399	3400	3401	3402	3403	3404	3405	3406	3407
D5_	3408	3409	3410	3411	3412	3413	3414	3415	3416	3417	3418	3419	3420	3421	3422	3423
D6_	3424	3425	3426	3427	3428	3429	3430	3431	3432	3433	3434	3435	3436	3437	3438	3439
D7_	3440	3441	3442	3443	3444	3445	3446	3447	3448	3449	3450	3451	3452	3453	3454	3455
D8_	3456	3457	3458	3459	3460	3461	3462	3463	3464	3465	3466	3467	3468	3469	3470	3471
D9_	3472	3473	3474	3475	3476	3477	3478	3479	3480	3481	3482	3483	3484	3485	3486	3487
DA_	3488	3489	3490	3491	3492	3493	3494	3495	3496	3497	3498	3499	3500	3501	3502	3503
DB_	3504	3505	3506	3507	3508	3509	3510	3511	3512	3513	3514	3515	3516	3517	3518	3519
DC_	3520	3521	3522	3523	3524	3525	3526	3527	3528	3529	3530	3531	3532	3533	3534	3535
DD_	3536	3537	3538	3539	3540	3541	3542	3543	3544	3545	3546	3547	3548	3549	3550	3551
DE_	3552	3553	3554	3555	3556	3557	3558	3559	3560	3561	3562	3563	3564	3565	3566	3567
DF_	3568	3569	3570	3571	3572	3573	3574	3575	3576	3577	3578	3579	3580	3581	3582	3583

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
E0_	3584	3585	3586	3587	3588	3589	3590	3591	3592	3593	3594	3595	3596	3597	3598	3599
E1_	3600	3601	3602	3603	3604	3605	3606	3607	3608	3609	3610	3611	3612	3613	3614	3615
E2_	3616	3617	3618	3619	3620	3621	3622	3623	3624	3625	3626	3627	3628	3629	3630	3631
E3_	3632	3633	3634	3635	3636	3637	3638	3639	3640	3641	3642	3643	3644	3645	3646	3647
E4_	3648	3649	3650	3651	3652	3653	3654	3655	3656	3657	3658	3659	3660	3661	3662	3663
E5_	3664	3665	3666	3667	3668	3669	3670	3671	3672	3673	3674	3675	3676	3677	3678	3679
E6_	3680	3681	3682	3683	3684	3685	3686	3687	3688	3689	3690	3691	3692	3693	3694	3695
E7_	3696	3697	3698	3699	3700	3701	3702	3703	3704	3705	3706	3707	3708	3709	3710	3711
E8_	3712	3713	3714	3715	3716	3717	3718	3719	3720	3721	3722	3723	3724	3725	3726	3727
E9_	3728	3729	3730	3731	3732	3733	3734	3735	3736	3737	3738	3739	3740	3741	3742	3743
EA_	3744	3745	3746	3747	3748	3749	3750	3751	3752	3753	3754	3755	3756	3757	3758	3759
EB_	3760	3761	3762	3763	3764	3765	3766	3767	3768	3769	3770	3771	3772	3773	3774	3775
EC_	3776	3777	3778	3779	3780	3781	3782	3783	3784	3785	3786	3787	3788	3789	3790	3791
ED_	3792	3793	3794	3795	3796	3797	3798	3799	3800	3801	3802	3803	3804	3805	3806	3807
EE_	3808	3809	3810	3811	3812	3813	3814	3815	3816	3817	3818	3819	3820	3821	3822	3823
EF_	3824	3825	3826	3827	3828	3829	3830	3831	3832	3833	3834	3835	3836	3837	3838	3839
F0_	3840	3841	3842	3843	3844	3845	3846	3847	3848	3849	3850	3851	3852	3853	3854	3855
F1_	3856	3857	3858	3859	3860	3861	3862	3863	3864	3865	3866	3867	3868	3869	3870	3871
F2_	3872	3873	3874	3875	3876	3877	3878	3879	3880	3881	3882	3883	3884	3885	3886	3887
F3_	3888	3889	3890	3891	3892	3893	3894	3895	3896	3897	3898	3899	3900	3901	3902	3903
F4_	3904	3905	3906	3907	3908	3909	3910	3911	3912	3913	3914	3915	3916	3917	3918	3919
F5_	3920	3921	3922	3923	3924	3925	3926	3927	3928	3929	3930	3931	3932	3933	3934	3935
F6_	3936	3937	3938	3939	3940	3941	3942	3943	3944	3945	3946	3947	3948	3949	3950	3951
F7_	3952	3953	3954	3955	3956	3957	3958	3959	3960	3961	3962	3963	3964	3965	3966	3967
F8_	3968	3969	3970	3971	3972	3973	3974	3975	3976	3977	3978	3979	3980	3981	3982	3983
F9_	3984	3985	3986	3987	3988	3989	3990	3991	3992	3993	3994	3995	3996	3997	3998	3999
FA_	4000	4001	4002	4003	4004	4005	4006	4007	4008	4009	4010	4011	4012	4013	4014	4015
FB_	4016	4017	4018	4019	4020	4021	4022	4023	4024	4025	4026	4027	4028	4029	4030	4031
FC_	4032	4033	4034	4035	4036	4037	4038	4039	4040	4041	4042	4043	4044	4045	4046	4047
FD_	4048	4049	4050	4051	4052	4053	4054	4055	4056	4057	4058	4059	4060	4061	4062	4063
FE_	4064	4065	4066	4067	4068	4069	4070	4071	4072	4073	4074	4075	4076	4077	4078	4079
FF_	4080	4081	4082	4083	4084	4085	4086	4087	4088	4089	4090	4091	4092	4093	4094	4095

Appendix B. Glossary

The following terms are defined as they are used in this manual. If you do not find the term you are looking for, refer to the *IBM Data Processing Glossary,* GC20-1699.

access arm: A part of a disk storage unit that is used to hold one or more reading and writing heads.

ALU: Arithmetic and logical unit.

base address: A given address from which a storage address is derived by combination with a relative address.

binary: (1) Pertaining to a characteristic or property involving a selection, choice, or condition in which there are two possibilities. (2) Pertaining to the numeration system with a radix of 2.

binary digit: In binary notation, either of the characters 0 or 1.

binary notation: A fixed radix notation where the radix is 2. For example, in binary notation the numeral 110.01 represents the number 1 x 2 squared plus 1 x 2 to the first power plus 1 x 2 to the minus 2 power, that is, 6-1/4.

binary number: Loosely, a binary numeral.

binary numeral: A binary representation of a number. For example, 101 is the binary numeral and V is the equivalent Roman numeral.

bit: (1) A binary digit. (2) Contraction of *binary digit,* the smallest unit of information in a binary system. A bit may be either a 1 (on) or a 0 (off).

blank: A code character to denote the presence of no information rather than the absence of information.

blank character: See *space character.*

block: A collection of contiguous records recorded as a unit. Blocks are separated by interrecord gaps on tape, and each block may contain one or more records.

bpi: The number of bits per inch per row (track) or the number of bytes per inch on tape. This term is used when referring to the recording density of the tape unit.

byte: A sequence of adjacent binary digits (8) operated upon as a unit.

card code: The combination of punched holes that represent characters (letters, digits, etc) in a punched card.

card column: A single line of punching positions parallel to the sort edge of the punched card.

card feed: A mechanism which moves cards into a machine one at a time.

card hopper: A device that holds cards and makes them available to a card feed mechanism.

card stacker: An output device that accumulates punched cards in a deck.

carry: One or more characters, produced in connection with an arithmetic operation on one digit place of two or more numerals in positional notation, that are forwarded to another digit place for processing there.

character: A letter, digit, or other symbol that is used as part of the organization, control, or representation of data.

character printer: A device that prints a single character at a time.

character set: An ordered set of unique representation called characters, for example, the 26 letters of the English alphabet, 0 and 1 of the Boolean alphabet, the set of signals in the Morse code alphabet.

check: A process for determining accuracy.

check bit: A binary check digit; for example, a parity bit.

check character: A character used for the purpose of performing a check.

COD: Change of direction latch.

code: A set of unambiguous rules specifying the way in which data may be represented.

code conversion: A process for changing the bit grouping for a character in one code into the corresponding bit grouping for a character in a second code.

collate: To compare and merge two or more similarly ordered sets of items into one ordered set.

collator: A device to collate sets of punched cards or other documents into a sequence.

column: A vertical arrangement of characters or other expressions. Loosely, a digit place.

command: An instruction.

comparison: The examination of the relationship between two similar items of data.

complement: A number that can be derived from a specified number by subtracting the specified number from another specified number.

conditional jump: A jump that occurs if specified criteria are met.

data: A representation of facts, concepts, or instructions in a formalized manner suitable for communication, interpretation, or processing by humans or automatic means.

data processing system: A network of machine component capable of accepting information, processing it according to a plan, and producing the desired results.

data set: Modem.

direct address: An address that specifies the location of an operand.

disk: A physical element of disk storage.

disk storage: A storage device which uses magnetic recording on flat rotating disks.

display: A visual presentation of data.

document: (1) A medium and the data recorded on it for human use; for example, a report sheet. (2) By extension, any record that has permanence and that can be read by man or machine.

edit: To modify the form or format of data, for example, to insert or delete characters such as page numbers or decimal points.

effective address: The address that is derived by applying any specified indexing or indirect addressing rules to the specified address and that is actually used to identify the current operand.

end-of-tape marker: A marker on a magnetic tape used to indicate the end of the permissible recording area; for example, a photo-reflective strip, a transparent section of tape, or a particular bit pattern.

end of transmission (EOT): The specific character or sequence of characters which indicates termination of sending.

EOT: End of transmission.

erase: To obliterate information from a storage medium.

erase head: A device on a magnetic tape unit whose sole function is to erase previous information before writing new information.

execute: To carry out an instruction or perform a routine.

fetch: To locate and load a quantity of data from storage.

field: In a record, a specified area used for a particular category of data, for example, a group of card columns used to represent a wage rate or a set of byte locations in a computer storage used to express another storage address.

file protection: Prevention of the destruction of data recorded on a volume by disabling the write head of a unit.

font: A family or assortment of characters of a given size and style.

format: A specific arrangement of data.

graphic: A symbol produced by a process such as handwriting, drawing, or printing.

graphic character: A character normally represented by a graphic.

halt instruction: A machine instruction which stops the execution of the program.

hard copy: A printed copy of machine output in a visually readable form, such as printed reports.

head: A device that reads, records, or erases data on a storage medium; for example, a small electromagnet used to read, write, or erase data on magnetic disk or tape.

hexadecimal: Pertaining to the numeration system with a radix of 16.

hit: A successful comparison of two items of data.

hopper: A card hopper.

I/O: Input/output. Input or output, or both.

indexed address: An address which is modified by the content of an index register prior to or during the execution of a computer instruction.

indexing: A technique of address modification often implemented by means of index registers.

indicator: A device which registers a condition in the computer.

indirect address: An address that specifies a storage location derived by adding an indexing factor to an index register.

initialize: To set counter, switches, and addresses to 0 or other starting values at the beginning of, or at a prescribed point in a computer routine.

initial program load (IPL): The procedure that causes the initial part of an operating system or other program to be loaded so that the program can then proceed under its own control.

input data: Data to be processed.

input device: A device used for conveying data to the processing unit.

input/output: (1) Commonly called I/O. (2) A general term for the equipment used to communicate with the processing unit.

instruction: A statement that specifies an operation and the values or locations of its operands.

instruction address: The address of the location where an instruction word is stored.

instruction format: The allocation of bits or bytes of a machine instruction to specific functions.

interblock gap: A blank space on magnetic tape that separates physical records.

interpreter: A device that prints on a punched card the data already punched in the card.

interrupt: To stop a process in such a way that it can be resumed.

jump: A departure from the normal sequence of executing instructions in a computer.

justification: The act of adjusting or arranging characters of digits to the left or right to fit a prescribed pattern.

justify: To align data about a specified reference.

line printer: A device that prints all characters of a line as a unit.

load: In programming, to enter data into storage or working registers.

loadpoint: The beginning of the usable portion of a reel of tape, indicated by a load point marker, where reading or writing is to begin.

location: Loosely, any place in which data can be stored.

loop: (n) A sequence of instructions that is repeated until a terminal condition exists. (v) To repeat an instruction or series of instructions until a terminal condition exists.

magnetic disk: A flat circular plate with a magnetic surface on which data can be stored by selective magnetization of portions of the flat surface.

magnetic tape: A tape with a magnetic surface on which data can be stored by selective polarization of portions of the surface.

main storage: The general purpose internal storage of a computer.

mask: A pattern of bits that is used to control the retention or elimination of portions of another pattern of bits.

mnemonic: Same as *mnemonic symbol.*

mnemonic symbol: A symbol chosen to assist the human memory; for example, the abbreviation MPY for multiply.

multivolume tape file: A file stored on more than one tape reel.

nines complement: The radix-minus 1 complement in decimal notation.

no-op: An instruction that performs no function except to proceed to the next instruction in sequence.

notation: A representational system which utilizes characters and symbols in positional relationships to express information.

NRZI (Non-Return-to-Zero IBM): A method of recording on tape where only the 1 bits are written as magnetized spots on tape.

number: A mathematical entity that may indicate quantity or amount of units.

one-address: Pertaining to an instruction format containing one address part.

operand: That which is operated upon. An operand is usually identified by an address part of an instruction.

operation: (1) A defined action, namely the act of obtaining a result from one or more operands in accordance with a rule that completely specifies the result for any permissible combination of operands. (2) The act specified by a single computer instruction.

operation code: A code that represents specified operations.

operator: A person who operates a machine.

output: The data that has been processed.

overflow: That portion of the result of an operation that exceeds the capacity of the intended unit of storage.

parity bit: A binary digit appended to an array of bits to make the sum of all the bits always odd or always even.

parity check: A check that tests whether the number of 1's or 0's in an array of binary digits is ood or even.

pass: One cycle of processing a body of data.

PE (phase encoding): A method of recording on tape where both 0- and 1-bits are written as magnetized spots. The 0- and 1-bit are opposite in polarity. This method allows distinction between 0-bits and no recording.

printer: A device which expresses coded characters as hard copy.

privileged instruction: An instruction which is executed only when the processing unit is in privileged state.

program: A series of actions proposed in order to achieve a certain result.

programmer: A person mainly involved in designing, writing, and testing programs.

programming: The design, the writing, and the testing of a program.

punched card: A card punched with a pattern of holes to represent data.

radix: In positional representation, the integral ration of the significances of any two specified adjacent digit positions.

radix-minus-1-complement: A complement obtained by subtracting each digit from 1 less than the radix.

read: To acquire or interpret data from a storage device, a data medium, or any other source.

read access time: The interval from issuance of a read forward read command given to the tape control when tape is not at load point, until the first data byte is read when tape is brought up to speed from stopped status.

reel: A mounting for a roll of tape.

register: A device capable of storing a specified amount of data, such as 2 bytes.

seek: To position the access mechanism of a disk drive at a specified track.

serdes: Serializer-deserializer. A device that changes data flow from parallel-by-bit to serial-by-bit or from serial-by-bit to parallel-by-bit.

space character: A normally nonprinting graphic character used to separate words.

storage: Pertaining to a device into which data can be entered, in which it can be held, and from which it can be retrieved at a later time.

storage capacity: The amount of data that can be contained in a storage device.

storage device: A device into which data can be inserted, in which it can be retained, and from which it can be retrieved.

subsystem: A secondary or subordinate system, usually capable of operating independently of or asynchronously with a controlling system.

tape labels: Special records at the beginning and ending of tape files. There are volume labels, and trailer labels. They are used to identify the reel and the recorded data file. They also contain certain housekeeping information.

tape mark: A special symbol that can be read from, or written on, magnetic tape. It is used to indicate the end of a file or a file segment, and to segregate the labels from data. Tape marks must be read in the same density in which they were written.

tape unit: A device containing a tape drive to move tape past the reading and writing heads, and containing the associated controls.

time-share: To use a device for two or more interleaved purposes.

time-sharing: Pertaining to the interleaved use of the time of a device.

track: The portion of a moving storage medium, such as a drum, tape, or disk, that is accessible to a given reading head position.

two-address: Pertaining to an instruction format containing two address parts.

verify: To determine whether a transcription of data or other operation has been accomplished correctly.

write: To record data in a storage device or a data medium.

write access time: The interval from the issuance of a write command given to the tape control, when the tape is not at load point, until the first data byte is written on tape when tape is brought up to speed from stopped status.

write-enable ring: A plastic ring that fits in a circular groove molded in the back (machine side) of the tape reel. This ring must be in place to enable the machine to write on the tape. When the ring is removed, only reading can take place; the file is protected from accidental writing, which could erase valuable information.

Index

International Business Machines Corporation

General Systems Division
4111 Northside Parkway N.W.
P.O. Box 2150
Atlanta, Georgia 30301
(U.S.A. only)

General Business Group/International
44 South Broadway
White Plains, New York 10601
U.S.A.
(International)

IBM System/3 Models 8, 10, 12, and 15 Components Reference Manual (File No. S3-01) Printed in U.S.A. GA21-9236-1

GA21-9236-1