SA21-9243-0

File No. S34-01

IBM System/34 Functions Reference

FUNCTIONS REFERENCE

IBM

SA21-9243-0

File No. S34-01

IBM System/34 Functions Reference

FUNCTIONS REFERENCE

Preface

This reference manual is intended for persons interested in the operation and characteristics of the System/34 at the machine code level. Readers should be familiar with data processing techniques and should understand programming at the machine code level.

This manual describes the machine instructions, status bytes, and other information needed to understand system programs from the hardware viewpoint.

Related Publications

- *IBM System/34 System Support Reference Manual,* SC21-5155
- *IBM System/34 Operator's Guide,* SC21-5158
- *IBM System/34 System Data Areas and Diagnostic Aids Handbook,* LY21-0049
- *IBM System/34 Displayed Messages Guide,* SC21-5159

First Edition (December 1977)

Changes are periodically made to the information herein; any such change will be included in a revision or technical newsletter to this manual.

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

A Reader's Comment Form is at the back of this publication. If the form has been removed, address your comments to IBM Corporation, Publications, Department 245, Rochester, Minnesota 55901. Comments become the property of IBM.

Contents

Abbreviations and Acronyms

A	address field
ac	alternating current
ACK	acknowledgment control character
Adv	advance
AID	attention identification
ALC	add logical characters
AL1	arithmetic logical 1
AL2	arithmetic logical 2
ARR	address recall register
ASCII	American National Standard Code for Information Interchange
AQE	allocation queue element
AZ	add zoned decimal
BC	branch on condition
BCC	block check character
bps	bits per second
BSC	binary synchronous communications
C	control field
CE	customer engineer
CHRNX	cylinder, head, record, record length, number of records
CLC	compare logical characters
CLI	compare logical immediate
COD	change-of-direction character
CRC	cyclic redundancy check
CSIPL	control storage initial program load
D	delete record
DISC	disconnect control character
EBCDIC	extended binary coded decimal interchange code
ENQ	enquiry control character
EOT	end of transmission control character
ETB	end of transmission block control character
ETX	end of text control character
F	flag
FIFO	first-in-first-out
H	head
I	information field
I/O	input/output
IOB	input/output block
IOCH	input/output control handler
IOS	input/output supervisor
IPL	initial program load
ITB	intermediate block control character
ITC	insert and test characters
IUS	intermediate block control character
JC	jump on condition
kbd	keyboard
L	load
LA	load address
LIFO	last-in-first-out
LMPR	load program mode register
LRC	longitudinal redundancy check
MIC	message identification code
MSIPL	main storage initial program load
MVC	move characters
MVI	move logical characters
MVX	move hexadecimal character
NAK	negative acknowledgment control character
NDM	normal disconnect mode
Nr	number received
NRM	normal response mode
NRZI	non-return to zero inverted
Ns	number sent
NSA	nonsequenced acknowledgment

P/F	poll/final
proc	processor
Pwr	power
R	record number
RNR	receive not ready
RR	receive ready
RVI	reverse interrupt control character
SCS	standard character string
SDLC	synchronous data link control
SLC	subtract logical characters
SNBU	switched network backup
SNRM	set normal response mode
SQS	system queue space
SS	sector number
SSP	System Support Program product
STX	start of text control character
SVC	supervisor call
SYN	synchronization control character

TBF	test bits off
TBN	test bits on
TCB	task control block
TTD	temporary text delay control character
TUB	terminal unit block
VRC	vertical redundancy check
WACK	wait for acknowledgment control character
WSIOCH	work station input/output control handler
WSQS	work station queue space
WSWA	work station work area
XFER	transfer
ZAZ	zero and add zoned

Chapter 1. Introduction

System/34 is a small, work station oriented data processing system. It works well in both batch and multiple-station environments. The system has a system console with a display screen and keyboard, and optionally, up to seven attached work stations. Each of these additional work stations may be either an additional keyboard and display or a tabletop serial printer. The display station is a 1920-character display and an attached keyboard. The tabletop printer is a serial matrix printer with printing speeds of 40, 80, or 120 characters per second.

The System/34 can also be configured with the following units:

- Line printer–160 or 300 lines per minute.
- Disk storage drive–one 8.6-megabyte drive, one 13.2-megabyte drive, or two drives totaling 27.1 megabytes.
- Diskette unit–diskette 1 drive or diskette 2D drive.

In addition, an optional feature permits attachment of a data communications line, either BSC or SDLC (SDLC available at version 3).

The system overlaps operations of the input/output devices with each other and with processing unit operations.

The models of System/34 differ in main storage capacity, diskette drive, and disk data storage capacities. Figure 1-1 lists the available System/34 models.

PARTS OF THE SYSTEM

Main Storage and Processor

Main storage minimum capacity is 32,768 (32K) 8-bit data bytes with either 49,153 (48K) or 65,536 (64K) bytes of main storage available.

The system unit, which has both a main storage processor and an integrated control processor, supplies all the arithmetic, logical, and input/output control functions for the system.

Display Station

The operator uses the display station to enter data to the system and communicate with the system. Each keyboard contains a set of alphameric keys (in the standard typewriter format), a set of adding machine keys in 10-key format, and a set of function keys the operator uses to select system functions. The display screen displays data and messages. Through programming, characters can be displayed on the display screen (and changed as needed) before they are sent to main storage. Under program control, main storage data and the contents of registers can be displayed and, if desired, changed by use of the keyboard.

Serial Printer

The serial printer is a bidirectional printer that prints at 40, 80, or 120 characters per second. This printer prints characters by a series of dots in a matrix; characters are made by printing a pattern of dots that matches a stored image in the printer adapter. The print line can be up to 132 characters long, and the character set contains 96 characters. In addition, the printer has a switch controlled carriage that permits printing of either six or eight lines per inch.

Line Printer

The line printer supplies fully buffered print rates of 160 lines per minute (Model 1) or 300 lines per minute (Model 2) with a 48-character set and 132 print positions. A 64- or 96-character set is also available. The printer has a switch controlled carriage that permits printing of either six or eight lines per inch.

Disk Storage

Each System/34 model has a disk storage drive with a disk that is not removable. Depending on the model, the disk storage is either 8,616,960, 13,271,040 bytes or 27,156,480 bytes.

System Model	Main Storage Capacity	Diskette	Disk Data Storage Capacity
A11	32,768 bytes	Diskette 1	8,616,960 bytes
A12	32,768 bytes	Diskette 1	13,271,040 bytes
A13	32,768 bytes	Diskette 1	27,156,480 bytes
A21	32,768 bytes	Diskette 2D	8,616,960 bytes
A22	32,768 bytes	Diskette 2D	13,271,040 bytes
A23	32,768 bytes	Diskette 2D	27,156,480 bytes
B11	49,152 bytes	Diskette 1	8,616,960 bytes
B12	49,152 bytes	Diskette 1	13,271,040 bytes
B13	49,152 bytes	Diskette 1	27,156,480 bytes
B21	49,152 bytes	Diskette 2D	8,616,960 bytes
B22	49,152 bytes	Diskette 2D	13,271,040 bytes
B23	49,152 bytes	Diskette 2D	27,156,480 bytes
C11	65,536 bytes	Diskette 1	8,616,960 bytes
C12	65,536 bytes	Diskette 1	13,271,040 bytes
C13	65,536 bytes	Diskette 1	27,156,480 bytes
C21	65,536 bytes	Diskette 2D	8,616,960 bytes
C22	65,536 bytes	Diskette 2D	13,271,040 bytes
C23	65,536 bytes	Diskette 2D	27,156,480 bytes

Figure 1-1. System Models

Diskette Drive

Each System/34 model has a diskette drive. The system uses either the IBM Diskette 1 or IBM Diskette 2D (or equivalent). This permits the system to read diskettes written by IBM 3741 Data Stations and similar devices and to exchange data with other systems. Data can also be written on diskettes and stored offline as backup data and programs. Data on diskettes that will not be used on other systems need not be in the basic data exchange format.

Data Communications Feature

Each model of the system can have, as a special feature, a communications adapter that supplies either binary synchronous communications (BSC) or the synchronous data link control (SDLC) (available at version 3). The BSC permits communication between System/34 and a remote system at data rates of from 600 to 9600 bits per second. SDLC permits communications between System/34 and a System/370 operating with the virtual telecommunications access method (VTAM). Data rates can be from 600 to 9600 bits per second.

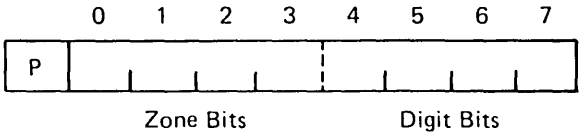
DATA FORMATS

Data in main storage is in 8-bit (plus parity) bytes. The instruction the system is executing determines how the data is used. A byte is used either as a character (decimal, alphabetic, or special) or as binary numbers (logical data).

The system uses EBCDIC (extended binary coded decimal interchange code) for storing and processing characters in main storage.

Character Format

In character format, each byte of data is divided into two groups of 4 bits each. Bits 0-3 make up the zone part, and bits 4-7 make up the digit part. The character format represents a decimal digit, a special control character, or one of the characters that can be printed or displayed by the system (these characters are *graphics*). The following shows the byte as interpreted for character format.



For decimal arithmetic operations, the zone bits of the rightmost byte in the field indicates the sign of the numbers. (The system ignores the zone bits in all other bytes during the operation.) Zones containing hex B or D (binary 1011 or 1101) specify a negative number. Any other hexadecimal digit in the zone specifies a positive number.

Binary Format (Logical Data)

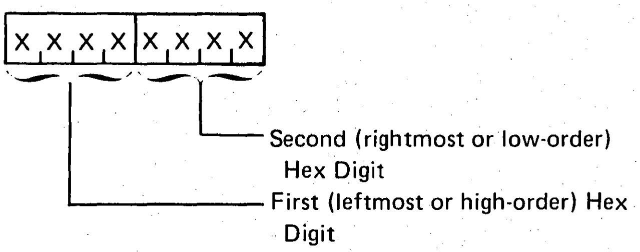
In binary format, bits in a byte define binary digits, and the complete byte is an unsigned binary number (a binary integer). Bits are said to be on if 1, off if 0. The following shows decimal 7 as a binary integer. Notice that the parity bit is set to 0 (see *Parity* later in this chapter).

P	0	1	2	3	4	5	6	7
0	0	0	0	0	0	1	1	1

Unsigned Binary Integer

Hexadecimal Notation

Each byte can be divided into two groups of 4 bits, and each of these groups can be represented as a single hexadecimal digit:


X X X X X X X X
Second (rightmost or low-order) Hex Digit
First (leftmost or high-order) Hex Digit

The hexadecimal value of each combination of binary bits is:

Binary Bits	Hex Digit	Binary Bits	Hex Digit
0000	0	1000	8
0001	1	1001	9
0010	2	1010	A
0011	3	1011	B
0100	4	1100	C
0101	5	1101	D
0110	6	1110	E
0111	7	1111	F

Throughout this manual, values stored in bytes are often shown in hexadecimal.

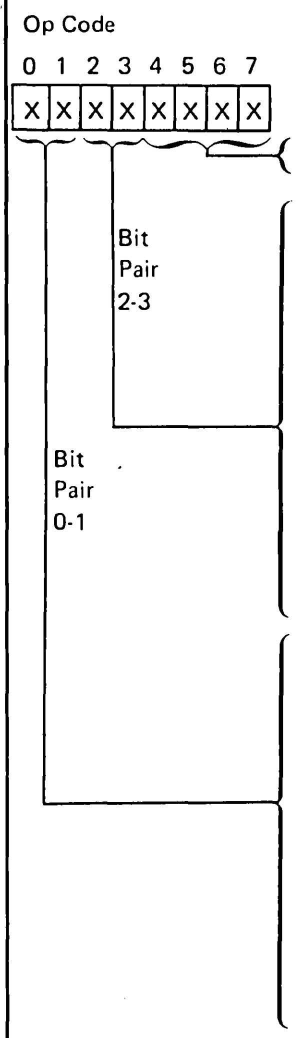
Parity

Each byte contains a parity bit that is generated by the system (and checked by the system during various operations). This bit ensures that the number of bits set to 1 in each byte is an odd number. (If the represented data causes the byte to have an even number of bits that are 1, the system sets the parity bit to 1 to make the byte contain an odd number of 1-bits. If the represented data has an odd number of bits, the system sets the parity bit to 0 to maintain an odd number of bits in the byte.)

ADDRESSING

Main storage is addressed in binary, using hexadecimal notation. Its locations are consecutively numbered from hex 0000 to the upper limit of storage. The location of any field or group of bytes is specified by the address of the rightmost (low-order or highest-numbered address) byte in the field. The exception is the insert and test character instruction, which specifies the leftmost byte.

A main storage address can be specified by either of two methods: direct addressing or base displacement addressing. The type of addressing to be used is specified by bits 0-3 of the first byte (the operation code) of the instruction. These 4 bits are looked at as pairs: bits 0-1 and bits 2-3. Bits 0 and 1 control addressing for operand 1. Bits 2 and 3 control addressing for operand 2. When bits 0-1 equal binary 11, operand 1 is not used; when bits 2-3 equal binary 11, operand 2 is not used. Figure 1-2 describes op code functions in addressing main storage.



Defines type of operation to be performed.

Specifies operand 2 main storage addressing activity for instruction:

00 – Operand 2 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 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 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 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 instruction and does not address main storage.

Figure 1-2. Op Code Function in Addressing Main Storage

Direct Addressing

When either or both bit pairs in the code is 00, the matching operand uses direct addressing.

When direct addressing is used, the storage address is taken directly from the instruction. The address in the instruction is 2 bytes long.

Base Displacement Addressing

When either or both bit pairs have one bit equal to 1 and the other bit equal to 0, the specified operand uses base displacement addressing.

In base displacement addressing, the contents of the 1-byte address in the instruction are added to the contents of a 2-byte address in an index register. The index register to be used is determined by the bit that is 1 (Figure 1-2). Both bit pairs can use the same index register during the execution of an instruction.

Any one value of an index register permits access to 255 storage positions.

INSTRUCTION FORMATS

Instruction formats are recognized by the way they address storage. The length of each instruction is determined by the type of addressing being performed.

All instruction formats have two parts in common: the op code and the Q-byte. Each of these parts is one byte long. The op code determines the type of addressing (therefore format of the instruction) and the operation to be performed. The function of the Q-byte is determined by the instruction and is described with each instruction.

Command Instructions

Command instructions are always 3 bytes long and all bits of the op code are 1's. In a command instruction, the Q-byte contains one of the following types of information, depending on the instruction:

Function specification

Jump condition

Op Code 1111	Q-Byte	Command

0 3
Bits

One-Address Instructions

One-address instructions can be either 3 or 4 bytes long. These instructions have either bit pair (bit 0-1 or bits 2-3) of the op code being both 1's. The other bit pair 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:

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

One-Address Instruction—Base-Displacement Addressing

Op Code 1110 1101 1011 0111	Q-Byte	Operand Displacement

0 3
Bits

One-Address Instruction—Direct Addressing

Op Code 0011 1100	Q-Byte	Operand (high-order byte of address)	Operand (low-order byte of address)

0 3
Bits

Two-Address Instructions

Two-address instructions can be 4, 5, or 6 bytes long. This instruction type is distinct in that *neither* bits 0-1 *nor* bits 2-3 of the op code are a pair of 1's. If all 4 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 1, one of the addresses is direct; the other address is base displacement, and the instruction is 5 bytes long. If 1 bit from each of the bit groups is 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 groups that is 1. If the bits equal 01, index register 1 is used; if the bits equal 10, index register 2 is used. Both addresses can use the same index register during one instruction.

Two-Address Instruction—Both Addresses Base Displacement

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

0 3
Bits

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 Address 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

MODES OF SYSTEM OPERATION

The system operates in four modes: burst, cycle steal, interrupt, and process.

Burst Mode

The system operates in *burst mode* while it moves data between main storage and the disk. In burst mode the system has a dedicated data path and, once data transfer starts, data moves quickly between the disk and main storage until all the specified data has been moved.

Cycle Steal Mode

Data may also be moved on a cycle steal basis. The attachment starts a request; the control storage processor addresses storage and controls the movement of data to and from the attachment, and to and from main and control storage.

Interrupt Mode

At the end of most input and output operations, the control processor is signaled that the operation has ended and that the program should branch to a special interrupt handler routine. While the system is processing data in the interrupt routine, it is said to be operating in the *interrupt mode*.

Process Mode

The system is free to handle normal I/O control and data processing operations when it is not operating in either the burst mode, interrupt mode, or cycle steal mode. At this time the system operates in *process mode*.

INSTRUCTION REGISTERS

Instruction Address Register (IAR)

The instruction address register holds the address of the first byte of the next sequential instruction in the stored program.

Address Recall Register (ARR)

The system places the next sequential address (that is, the address of the instruction that follows the branch on condition instruction) in the address recall register when the program branches. At the end of the branched to routine, the program can load the contents of the address recall register into the instruction address register; this returns the program to the point at which the branch occurred.

The address recall register is also affected by zero and add zoned, load register, add to register, decimal add and subtract, and insert and test characters instructions. (All instructions are described in Chapter 3.)

Index Registers 1 and 2 (XR1 and XR2)

Index registers hold base addresses for base displacement addressing.

Op Register

The op register holds each control word as it is fetched from main storage. Control words are used for hardware functions and selections, setting of the program status register, selection of the index registers, and CPU clock controls.

Q Register

The Q register holds a byte that specifies the length of the operands used in ALU operations. This length count is decreased as the instruction is executed. It is also used with the op register to control operations and to select registers to be changed or stored.

Program Mode Register

The program mode register controls main storage address translation and protection. Control storage instructions are used to load or sense the program mode register. The program mode register can also be loaded from the main storage processor using the load program mode register instruction. Bit assignments in the 8-bit program mode register are as follows:

Bit	Meaning When On
0	Dispatching disabled
1	Not used
2	Not used
3	Not used
4	Main storage processor instruction address register is translated
5	Main storage processor operand 2 addresses are translated
6	Main storage processor operand 2 addresses are translated
7	Nonprivileged mode

Address Translation Registers (ATRs)

Address translation registers (ATRs) provide main storage address translation capability by 2K address blocks. Sixty-four local storage registers named address translation registers (ATR) provide the address translation function. Thirty-two of these are for program level (task) address translation; the other 32 are for input/output uses.

Each ATR stores 1 byte of data. Address translation register data contents of hex 00 through hex 1F provide address translation by addressing 32 pages (or 2K bytes) in main storage. Storage protection for a 2K address block is provided by loading its address translation register with hex FF. The storage protection mechanism is operable only when address translation is in effect. Any attempt to access a protected storage location causes a program check.

Translate mode is controlled through the program mode register or control mode register contents. When in translate mode, the program mode register or control mode register direct main storage address register (MSAR), bits 0 through 4, to select one of 64 address translation registers. The contents of the address translation register then perform the addressing function of main storage address register bits 0 through 4. When not in translate mode, main storage address register bits 0 through 4 directly control main storage addressing.

INPUT/OUTPUT BLOCKS (IOBs)

Each input and output function has specific parameters that the program must define before the operation is performed. The parameters are moved into input/output blocks, which are consecutive main storage positions into which parameters are placed in defined fields.

When an input/output operation is started, the program must present the address of the leftmost byte of the input/output block to the system (in index register 1).

When an input/output block is needed for a function, this manual describes that input/output block in the chapter that describes the function.

GENERAL INPUT/OUTPUT OPERATIONS

All input/output operations are done by the input/output task for that operation. At initialization time, the control storage code for the I/O device is loaded and the attachment is enabled.

Chapter 2. Control Panels

OPERATOR PANEL

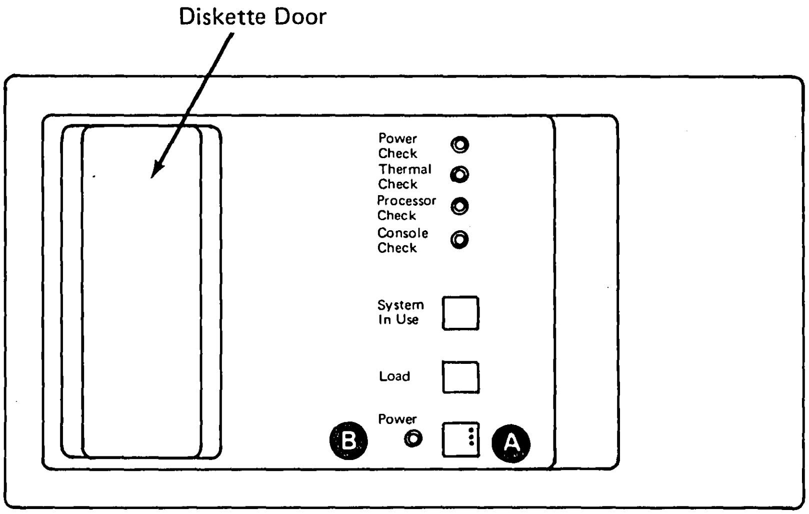


Figure 2-1. System/34 Operator Panel

Power Switch Ⓐ

Set the Power switch to I to power on the system. If the keylock feature is installed, turn the key to the horizontal (on) position. When you switch the power on, a system reset occurs and the Power light comes on.

Set the Power switch to O to power off the system. If the Keylock feature is installed, turn the key to the vertical (off) position. When you switch the power off:

- The Power light goes off.
- The contents of registers and storage are lost.
- Information stored in the power failure latches about the most recent power failure is kept.

Power Light Ⓑ

The Power light is on when system power is on. The Power light is off when system power is off.

Load Switch/Light

Press the Load switch to start the control storage initial program load and main storage initial program load sequences.

When you press the Load switch, the Load light comes on. The Load light remains on until the first part of the control storage routine is loaded correctly.

System In Use Light

The System In Use light comes on when one or more programs or commands are active in main storage. The System In Use light goes off when no programs or commands are using main storage. When on, the System In Use light indicates that programs have not completed running, so the system should *not* be powered off, and the Load switch should *not* be pressed.

Power Check Light

The Power Check light comes on if the voltage or current in one of the power supplies does not meet specifications. When the Power Check light comes on, the system is powered off, but information stored in the power failure latches is kept. If this light is on, notify the IBM customer engineer.

Thermal Check Light

If one of the system thermal sensors detects an overheated condition, system power automatically turns off and the Thermal Check light comes on. The light remains on until the overheated condition is corrected and the Power switch is turned off. Power can then be restored to the system by turning the Power switch on.

Processor Check Light

The Processor Check light comes on if the processing unit senses an error for which there is no correction procedure. If the Processor Check light comes on, press the Load switch to start a new initial program load sequence.

Console Check Light

The Console Check light comes on if the system console or the work station controller fails. If the system console fails, another work station can be assigned as the system console before processing continues (if an alternate is specified in the system configuration). If the work station controller fails, the cause of the failure must be found and corrected before processing continues. The Console Check light goes off after the cause of the failure is corrected.

IMMEDIATE POWER OFF SWITCH

The Immediate Power Off switch, on the left side of the system unit:

- Must remain set to I (on) during normal system operation.
- When set to O (off), removes all system power except AC to the I/O control supply.

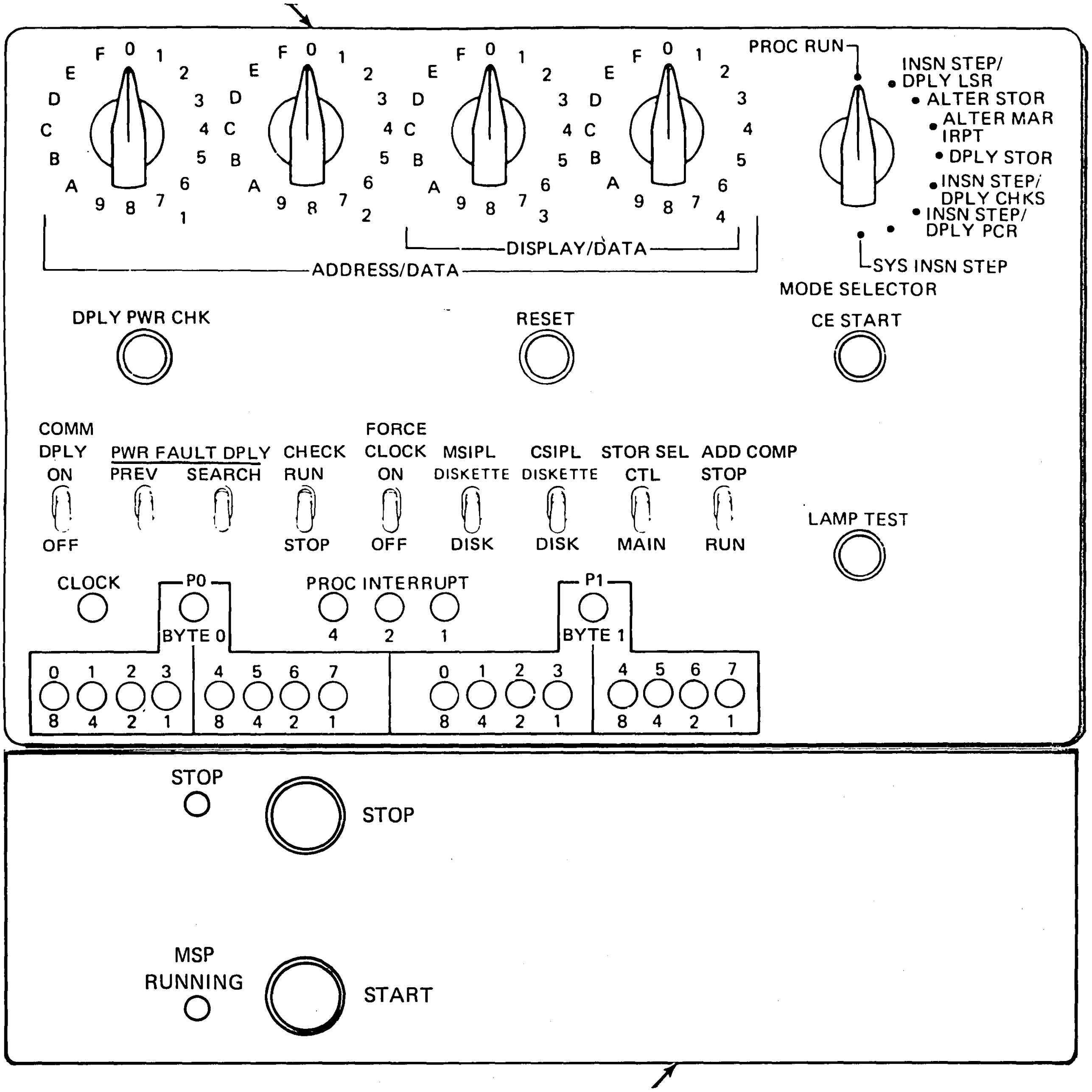
CAUTION
The Immediate Power Off switch is for emergency use only. Do not use the Immediate Power Off switch to power on and power off the system. When powering on, you must use the Power switch on the operator panel to initialize the system correctly.

CE PANEL AND SUBPANEL

Although this is called a CE panel, some of the switches on the panel are used by the operator and the programmer. When these switches are needed, the specific application which needs them instructs you on their correct use.

The Dply Pwr Chk switch, the Reset switch, and the CE Start switch are for the use of the customer engineer. The other switches are described in this section.

CE Panel



CE Subpanel

Start Switch

The Start switch is on the CE subpanel, below the CE panel. When the Stop light is on, the alter/display routine is usually in control and the start switch has no function. It only causes the Stop light to go off.

MSP Running Light

The MSP Running light is on the CE subpanel, below the CE panel. The MSP Running light comes on if the Start switch is pressed and the system can execute programs. The MSP Running light remains on as long as the main storage processor clock is running. The MSP Running light goes off when the main storage processor clock is stopped.

Stop Switch

The Stop switch is on the CE subpanel, below the CE panel. After each system instruction is executed, the control storage routine tests to see if the Stop switch was pressed. If the Stop switch was pressed:

- The main storage processor stops.
- The control processor continues to run.
- The alter/display routine becomes active and the option menu shows on the system console.

Stop Light

The Stop light is on the CE subpanel, below the CE panel. The Stop light comes on:

- When you press the Stop switch.
- When the system has been powered on.
- If an address compare stop occurs for a main storage address.

The Stop light goes off when you press the Load switch or the Start switch.

Address/Data and Display/Data Switches

These four 16-position rotary switches are used in conjunction with other switches on the CE panel to enter, alter, or display data stored in main storage or local storage registers. During normal operation, these switches are set to the 0 position.

Mode Selector Switch

During normal processing operations, the Mode Selector switch must be set to the Proc Run position. All other positions are associated with diagnostic procedures used by persons servicing your system.

Toggle Switches

Except for the Comm Dply (communications display) switch, the toggle switches on the CE panel are to be set to the down position for normal operation. The programmer or the operator may need to use these switches for certain procedures. When this is necessary, detailed instructions are given.

The Comm Dply switch is present on the CE panel only if a communications adapter is installed on the system. When set to the On position, the Comm Dply switch activates the leftmost six lights at the bottom of the CE panel. The lights indicate the status of the communications interface lines as shown in the following figure.

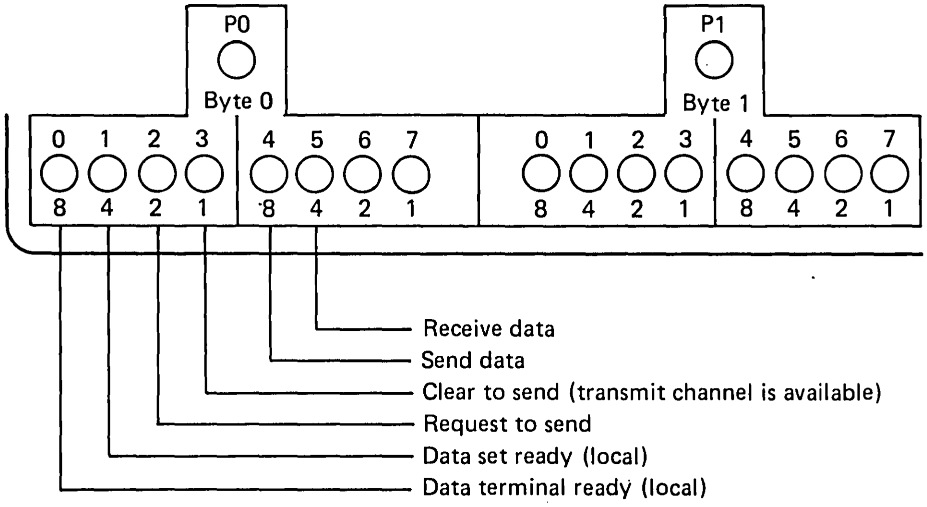


Figure 2-2. Communication Display Indicators

Display Lights

The group of lights at the bottom of the CE panel displays the contents of certain system registers and presents system status information. The customer engineer uses the Mode Selector switch to select the type of information to be displayed.

Lamp Test Switch

When you press the Lamp Test switch:

- If system power is on, all system lights come on.
- If system power is off, but the circuit breaker (CB1) is on:
  - The Power Check and Thermal Check lights on the operator panel come on.
  - The display byte 0 lights on the CE panel come on.

Chapter 3. Machine Instructions

Each System/34 machine instruction is described here in detail. The instructions are in three groups:

- Arithmetic
- Data handling
- Logical

Arithmetic Machine Instructions

ZERO AND ADD ZONED (ZAZ)

Operands	Op Code (hex)	Q-Byte[1] (hex)		Operand Addresses[2] (hex)			
	Byte 1	Byte 2		Byte 3	Byte 4	Byte 5	Byte 6
A1(L1),A2(L2)	04	L1-L2	L2-1	Operand 1 address		Operand 2 address	
A1(L1),D2(L2,R1)	14	L1-L2	L2-1	Operand 1 address		Op 2 disp from XR1	
A1(L1),D2(L2,R2)	24	L1-L2	L2-1	Operand 1 address		Op 2 disp from XR2	
D1(L1,R1),A2(L2)	44	L1-L2	L2-1	Op 1 disp from XR1	Operand 2 address		
D1(L1,R1),D2(L2,R1)	54	L1-L2	L2-1	Op 1 disp from XR1	Op 2 disp from XR1		
D1(L1,R1),D2(L2,R2)	64	L1-L2	L2-1	Op 1 disp from XR1	Op 2 disp from XR2		
D1(L1,R2),A2(L2)	84	L1-L2	L2-1	Op 1 disp from XR2	Operand 2 address		
D1(L1,R2),D2(L2,R1)	94	L1-L2	L2-1	Op 1 disp from XR2	Op 2 disp from XR1		
D1(L1,R2),D2(L2,R2)	A4	L1-L2	L2-1	Op 1 disp from XR2	Op 2 disp from XR2		

[1] The Q-byte designates the operand length:
L1-L2 (4 bits) = the number of bytes in operand 1, minus the number of bytes in operand 2.
L2-1 (4 bits) = the 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 machine instruction copies 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 main storage processor fills the extra positions with high-order EBCDIC zeros (hex F0).

The main storage processor 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 moved is either zero or positive, or (2) hex D (binary 1101) if the value moved is negative.

Program Notes

- The second operand is not changed unless the fields overlap.

- The system stores the rightmost address of operand 1 in the address recall register (ARR) if not recomplemented and the rightmost address minus 1 if recomplemented.

CAUTION
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.

Resulting Program Status Byte Settings

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	← Storage Positions

Operand 2 before and after Operation

			F4	F2	F5	
			001E	001F	0020	← Storage Positions

Operand 1 after Operation

	F0	F0	F4	F2	F5	
	000C	000D	000E	000F	0010	← Storage Positions

Program Status Register before Operation

00000001

0 7 ← Bits

Program Status Register after Operation

00000100

0 7 ← Bits

ADD ZONED DECIMAL (AZ)

Operands	Op Code (hex)	Q-Byte[1] (hex)		Operand Addresses[2] (hex)			
	Byte 1	Byte 2		Byte 3	Byte 4	Byte 5	Byte 6
A1(L1),A2(L2)	06	L1-L2	L2-1	Operand 1 address		Operand 2 address	
A1(L1),D2(L2,R1)	16	L1-L2	L2-1	Operand 1 address		Op 2 disp from XR1	
A1(L1),D2(L2,R2)	26	L1-L2	L2-1	Operand 1 address		Op 2 disp from XR2	
D1(L1,R1),A2(L2)	46	L1-L2	L2-1	Op 1 disp from XR1	Operand 2 address		
D1(L1,R1),D2(L2,R1)	56	L1-L2	L2-1	Op 1 disp from XR1	Op 2 disp from XR1		
D1(L1,R1),D2(L2,R2)	66	L1-L2	L2-1	Op 1 disp from XR1	Op 2 disp from XR2		
D1(L1,R2),A2(L2)	86	L1-L2	L2-1	Op 1 disp from XR2	Operand 2 address		
D1(L1,R2),D2(L2,R1)	96	L1-L2	L2-1	Op 1 disp from XR2	Op 2 disp from XR1		
D1(L1,R2),D2(L2,R2)	A6	L1-L2	L2-1	Op 1 disp from XR2	Op 2 disp from XR2		

[1] The Q-byte designates the operand length:
L1-L2 (4 bits) = the number of bytes in operand 1, minus the number of bytes in operand 2.
L2-1 (4 bits) = the 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 machine instruction algebraically adds the second operand to the first operand and stores the result in the first operand.

The main storage processor 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 (binary 1101) if the result is negative.

Program Notes

CAUTION
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.

- The second operand is not changed unless the fields overlap.

- The system does not check for valid decimal digits in either operand.

- The decimal overflow condition indicator (program status bit 4), 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 stores the rightmost address of operand 1 in the address recall register if not recomplemented, or the rightmost address minus 1 if recomplemented and the result is zero, or the leftmost address minus 1 if recomplemented and the result is not zero.

Resulting Program Status Byte Settings

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	← Storage Positions

Operand 2 Before and After Operation

			F4	F2	F5	
			001E	001F	0020	← Storage Positions

Operand 1 After Operation

	F7	F6	F7	F9	F4	
	000C	000D	000E	000F	0010	← Storage Positions

Program Status Register Before Operation

00000001

0 7 ← Bits

Program Status Register After Operation

00000100

0 7 ← Bits

SUBTRACT ZONED DECIMAL (SZ)

Operands	Op Code (hex)	Q-Byte[1] (hex)		Operand Addresses[2] (hex)			
	Byte 1	Byte 2		Byte 3	Byte 4	Byte 5	Byte 6
A1(L1),A2(L2)	07	L1-L2	L2-1	Operand 1 address		Operand 2 address	
A1(L1),D2(L2,R1)	17	L1-L2	L2-1	Operand 1 address		Op 2 disp from XR1	
A1(L1),D2(L2,R2)	27	L1-L2	L2-1	Operand 1 address		Op 2 disp from XR2	
D1(L1,R1),A2(L2)	47	L1-L2	L2-1	Op 1 disp from XR1	Operand 2 address		
D1(L1,R1),D2(L2,R1)	57	L1-L2	L2-1	Op 1 disp from XR1	Op 2 disp from XR1		
D1(L1,R1),D2(L2,R2)	67	L1-L2	L2-1	Op 1 disp from XR1	Op 2 disp from XR2		
D1(L1,R2),A2(L2)	87	L1-L2	L2-1	Op 1 disp from XR2	Operand 2 address		
D1(L1,R2),D2(L2,R1)	97	L1-L2	L2-1	Op 1 disp from XR2	Op 2 disp from XR1		
D1(L1,R2),D2(L2,R2)	A7	L1-L2	L2-1	Op 1 disp from XR2	Op 2 disp from XR2		

[1] The Q-byte designates the operand length:
L1-L2 (4 bits) = the number of bytes in operand 1, minus the number of bytes in operand 2.
L2-1 (4 bits) = the 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 machine instruction algebraically subtracts operand 2 from operand 1, byte by byte, and stores the result in operand 1. The main storage processor 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

CAUTION

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.

- The second operand is not changed unless the fields overlap.

- The system does not check for valid decimal digits in either operand.

- The decimal overflow condition indicator (program status bit 4), which may be set during this operation, can be 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 stores the rightmost address of operand 1 in the address recall register if not recomplemented, or the rightmost address minus 1 if recomplemented and the result is zero, or the leftmost address minus 1 if recomplemented and the result is not zero.

Resulting Program Status Byte Settings

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	← Storage Positions

Operand 2 before and after Operation

			F4	F2	F5	
			001E	001F	0020	← Storage Positions

Operand 1 after Operation

	F7	F5	F9	F4	F4	
	000C	000D	000E	000F	0010	← Storage Positions

Program Status Register before Operation

00000001	
0	7 ← Bits

Program Status Register after Operation

00000100	
0	7 ← Bits

ADD LOGICAL CHARACTERS (ALC)

Operands	Op Code (hex)	Q-Byte[1] (hex)	Operand Addresses[2] (hex)			
	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
A1(L1),A2	0E	L1-1	Operand 1 address		Operand 2 address	
A1(L1),D2(,R1)	1E	L1-1	Operand 1 address		Op 2 disp from XR1	
A1(L1),D2(,R2)	2E	L1-1	Operand 1 address		Op 2 disp from XR2	
D1(L1,R1),A2	4E	L1-1	Op 1 disp from XR1	Operand 2 address		
D1(L1,R1),D2(,R1)	5E	L1-1	Op 1 disp from XR1	Op 2 disp from XR1		
D1(L1,R1),D2(,R2)	6E	L1-1	Op 1 disp from XR1	Op 2 disp from XR2		
D1(L1,R2),A2	8E	L1-1	Op 1 disp from XR2	Operand 2 address		
D1(L1,R2),D2(,R1)	9E	L1-1	Op 1 disp from XR2	Op 2 disp from XR1		
D1(L1,R2),D2(,R2)	AE	L1-1	Op 1 disp from XR2	Op 2 disp from XR2		

[1] The Q-byte designates the operand length:
L1-1 = the number of bytes in either operand, 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 machine instruction adds the binary number in operand 2 to the binary number in operand 1 and stores the result in operand 1.

Program Note

CAUTION
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.

- The system resets the binary overflow bit during this operation if a carry does not occur from the high-order byte.

Resulting Program Status Byte Settings

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

Note: Index register 1 = 0CC0

Operand 1 before Operation

		00110101	11001011	11101101	01100100	
		0CBD	0CBE	0CBF	0CC0	← Storage Positions

Operand 2 before and after Operation

		01011011	01010101	01111000	11001101	
		0CCD	0CCE	0CCF	0CD0	← Storage Positions

Operand 1 after Operation

		10010001	00100001	01100110	00110001	
		0CBD	0CBE	0CBF	0CC0	← Storage Positions

Program Status Register before Operation

00000001

0 7 ← Bits

Program Status Register after Operation

00000010

0 7 ← Bits

SUBTRACT LOGICAL CHARACTERS (SLC)

Operands	Op Code (hex)	Q-Byte[1] (hex)	Operand Addresses[2] (hex)			
	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
A1(L1),A2	0F	L1-1	Operand 1 address		Operand 2 address	
A1(L1),D2(,R1)	1F	L1-1	Operand 1 address		Op 2 disp from XR1	
A1(L1),D2(,R2)	2F	L1-1	Operand 1 address		Op 2 disp from XR2	
D1(L1,R1),A2	4F	L1-1	Op 1 disp from XR1	Operand 2 address		
D1(L1,R1),D2(,R1)	5F	L1-1	Op 1 disp from XR1	Op 2 disp from XR1		
D1(L1,R1),D2(,R2)	6F	L1-1	Op 1 disp from XR1	Op 2 disp from XR2		
D1(L1,R2),A2	8F	L1-1	Op 1 disp from XR2	Operand 2 address		
D1(L1,R2),D2(,R1)	9F	L1-1	Op 1 disp from XR2	Op 2 disp from XR1		
D1(L1,R2),D2(,R2)	AF	L1-1	Op 1 disp from XR2	Op 2 disp from XR2		

[1] The Q-byte designates the operand length:
L1-1 = the number of bytes in either operand, 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 machine instruction subtracts the binary number in operand 2 from the binary number in operand 1 and stores the result in operand 1. If the second operand is numerically larger than the number stored in the first operand, the result occurs as if 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

CAUTION
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.

Resulting Program Status Byte Settings

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

Example

Instruction

AF	03	00	10

Note: Index register 2 = 0CC0

Operand 1 before Operation

		10010110	01011010	01110111	10111111	
		0CBD	0CBE	0CBF	0CC0	← Storage Positions

Operand 2 before and after Operation

		01110100	10000110	01100010	10100100	
		0CCD	0CCE	0CCF	0CD0	← Storage Positions

Operand 1 after Operation

		00100001	11010100	00010101	00011011	
		0CBD	0CBE	0CBF	0CC0	← Storage Positions

Program Status Register before Operation

00000001

0 7←Bits

Program Status Register after Operation

00000100

0 7←Bits

ADD TO REGISTER (A)

Operands	Op Code (hex)	Q-Byte[1] (binary)	Operand Address[2] (hex)	
	Byte 1	Byte 2	Byte 3	Byte 4
A1,RX	36	Rx	Operand 1 address	
D1(,R1),RX	76	Rx	Op 1 disp from XR1	
D1(,R2),RX	B6	Rx	Op 1 disp from XR2	

[1] Rx specifies the register whose contents are modified by the machine instruction.

[2] Operand 1 is a 2-byte field addressed by its rightmost byte; operand 2 is not used.

Operation

This machine instruction adds the binary number in operand 1 to the contents of the 2-byte register selected by the Q-byte and stores the result in the register. The Q-byte coding is:

Q-Byte Binary	Hex	Register Specified
0000 0000	00	None. The system ignores (no-ops) the instruction.
0000 0001	01	XR1.
0000 0010	02	XR2.
0000 0100	04	Program status register.
0000 1000	08	Address recall register.
0001 0000	10	Instruction address register.
0010 0000	20	Instruction address register.
0100 0000	40	Reserved; do not use.
1000 0000	80	Reserved; do not use.

Program Notes

- This instruction changes the contents of only one register at a time.
- This machine instruction does not change the operand.
- Subtract from the register by placing, in the operand, the twos complement of the number to be subtracted.
- Adding to the program status register causes unpredictable results; a hex 04 is forced into the high byte before the addition is done.

Resulting Program Status Byte Settings

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	← Storage Positions

Index Register 2 before Operation

00110101	01101010

Index Register 2 after Operation

01111101	10001010

Program Status Byte after Operation

00000010

0 7 ← Bits

Data Handling Machine Instructions

MOVE HEXADECIMAL CHARACTER (MVX)

Operands	Op Code (hex)	Q-Byte[1] (hex)	Operand Addresses[2] (hex)			
	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
A1(I),A2	08	I	Operand 1 address		Operand 2 address	
A1(I),D2(,R1)	18	I	Operand 1 address		Op 2 disp from XR1	
A1(I),D2(,R2)	28	I	Operand 1 address		Op 2 disp from XR2	
D1(I,R1),A2	48	I	Op 1 disp from XR1	Operand 2 address		
D1(I,R1),D2(,R1)	58	I	Op 1 disp from XR1	Op 2 disp from XR1		
D1(I,R1),D2(,R2)	68	I	Op 1 disp from XR1	Op 2 disp from XR2		
D1(I,R2),A2	88	I	Op 1 disp from XR2	Operand 2 address		
D1(I,R2),D2(,R1)	98	I	Op 1 disp from XR2	Op 2 disp from XR1		
D1(I,R2),D2(,R2)	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 1-byte operand is used in the operation.

[2] Both operands are 1-byte fields.

Operation

This machine instruction moves the numeric part (bits 4-7) or the zone part (bits 0-3) of the second operand to the numeric or zone part 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 bits in the Q-byte immediate data should be 0's.
- The second operand is not changed unless both operands specify the same byte.

Resulting Program Status Byte Settings

This machine instruction does not affect the program status register.

Example

Instruction

98	01	A0	65

Index register 1 = 2B15
Index register 2 = 1F20

Operand 1 before Operation

2F

1FC0 ← Storage Position

Operand 2 before and after Operation

4C

2B7A ← Storage Position

Operand 1 after Operation

CF

1FC0 ← Storage Position

MOVE CHARACTERS (MVC)

Operands	Op Code (hex)	Q-Byte[1] (hex)	Operand Addresses[2] (hex)			
	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
A1(L1),A2	0C	L-1	Operand 1 address		Operand 2 address	
A1(L1),D2(,R1)	1C	L-1	Operand 1 address		Op 2 disp from XR1	
A1(L1),D2(,R2)	2C	L-1	Operand 1 address		Op 2 disp from XR2	
D1(L1,R1),A2	4C	L-1	Op 1 disp from XR1	Operand 2 address		
D1(L1,R1),D2(,R2)	5C	L-1	Op 1 disp from XR1	Op 2 disp from XR1		
D1(L1,R1),D2(,R2)	6C	L-1	Op 1 disp from XR1	Op 2 disp from XR2		
D1(L1,R2),A2	8C	L-1	Op 1 disp from XR2	Operand 2 address		
D1(L1,R2),D2(,R1)	9C	L-1	Op 1 disp from XR2	Op 2 disp from XR1		
D1(L1,R2),D2(,R2)	AC	L-1	Op 1 disp from XR2	Op 2 disp from XR2		

[1] The Q-byte designates the operand length:
L-1 = the number of bytes in either operand, 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 machine instruction places the contents of operand 2, byte by byte, into operand 1. It is possible to propagate one character through a complete field by setting the operand 2 address one byte to the right of the operand 1 address.

Program Note

CAUTION
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. The second operand is not changed unless the fields overlap.

Resulting Program Status Byte Settings

This machine instruction does not affect the program status register.

Example

Instruction

0C	05	1A	06	2B	5A

Operand 1 before Operation

D1	C1	D4	C5	E2		
1A01	1A02	1A03	1A04	1A05	1A06	← Storage Positions

Operand 2 before Operation

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

Operand 1 after Operation

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

EDIT (ED)

Operands	Op Code (hex)	Q-Byte[1] (hex)	Operand Addresses[2] (hex)			
	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
A1(L1),A2	0A	L1-1	Operand 1 address		Operand 2 address	
A1(L1),D2(,R1)	1A	L1-1	Operand 1 address		Op 2 disp from XR1	
A1(L1),D2(,R2)	2A	L1-1	Operand 1 address		Op 2 disp from XR2	
D1(L1,R1),A2	4A	L1-1	Op 1 disp from XR1	Operand 2 address		
D1(L1,R1),D2(,R1)	5A	L1-1	Op 1 disp from XR1	Op 2 disp from XR1		
D1(L1,R1),D2(,R2)	6A	L1-1	Op 1 disp from XR1	Op 2 disp from XR2		
D1(L1,R2),A2	8A	L1-1	Op 1 disp from XR2	Operand 2 address		
D1(L1,R2),D2(,R1)	9A	L1-1	Op 1 disp from XR2	Op 2 disp from XR1		
D1(L1,R2),D2(,R2)	AA	L1-1	Op 1 disp from XR2	Op 2 disp from XR2		

[1] The Q-byte designates the operand length:
L1-1 = the number of bytes in either operand, minus 1.
Operand 2 must contain as many bytes as there are hex 20s in operand 1.

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

Operation

This machine instruction replaces bytes containing hex 20 in operand 1 with characters from operand 2. Starting at the rightmost position in both operands, the processing unit inspects operand 1 for hex 20s. When the system finds the first hex 20, it moves the first byte from operand 2 into that hex 20 location, then inspects the following bytes in operand 1 for the next sequential hex 20. Locating the second hex 20, the system moves the second byte from operand 2 into that operand 1 position. The operation continues until the last byte in operand 1 has been examined for hex 20. During the operation, the system sets the zone bits of all replaced operand 1 bytes to hex F (binary 1111).

Program Note

Operand 2 is not changed during this instruction.

Resulting Program Status Byte Settings

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	.	20	20	ƀ	*
00B5	00B6	00B7	00B8	00B9	00BA	00BB	00BC	00BD	00BE	00BF

← Storage Positions

Operand 2 before and after Operation

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

← Storage Positions

Note: R represents hex D9 (-9)

Operand 1 after Operation

$	0	,	1	0	8	.	0	9	ƀ	*
00B5	00B6	00B7	00B8	00B9	00BA	00BB	00BC	00BD	00BE	00BF

← Storage Positions

Note: Storage position 00BD contains a 9 because the zone bits of all replaced characters in the edit pattern are set to hex F (binary 1111).

Program Status Byte after Operation

00000010

0 7 ← Bits

INSERT AND TEST CHARACTERS (ITC)

Operands	Op Code (hex)	Q-Byte[1] (hex)	Operand Addresses[2] (hex)			
	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
A1(L1),A2	0B	L1-1	Operand 1 address		Operand 2 address	
A1(L1),D2(,R1)	1B	L1-1	Operand 1 address		Op 2 disp from XR1	
A1(L1),D2(,R2)	2B	L1-1	Operand 1 address		Op 2 disp from XR2	
D1(L1,R1),A2	4B	L1-1	Op 1 disp from XR1	Operand 2 address		
D1(L1,R1),D2(,R1)	5B	L1-1	Op 1 disp from XR1	Op 2 disp from XR1		
D1(L1,R1),D2(,R2)	6B	L1-1	Op 1 disp from XR1	Op 2 disp from XR2		
D1(L1,R2),A2	8B	L1-1	Op 1 disp from XR2	Operand 2 address		
D1(L1,R2),D2(,R1)	9B	L1-1	Op 1 disp from XR2	Op 2 disp from XR1		
D1(L1,R2),D2(,R2)	AB	L1-1	Op 1 disp from XR2	Op 2 disp from XR2		

[1] The Q-byte designates the operand length:
L1-1 = the number of bytes in either operand, minus 1.
Operand 2 is a 1-byte field.

[2] Address operand 1 by its leftmost position.

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.

If the leftmost byte of a field to be printed contains a character that must not be replaced (for example, a dollar sign), the first operand should start with the byte to the right of that character.

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

Program Notes

- Operand 2 is not changed.

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

Resulting Program Status Byte Settings

This machine instruction does not affect the program status register.

Example

Instruction

0B	09	00	B6	00	10

Operand 1 before Operation

$	0	,	1	0	8	.	0	9	ƀ	*	
00B5	00B6	00B7	00B8	00B9	00BA	00BB	00BC	00BD	00BE	00BF	←Storage Positions

Operand 2 before and after Operation

*	
0010	←Storage Position

Operand 1 after Operation

$	*	*	1	0	8	.	0	9	ƀ	*	
00B5	00B6	00B7	00B8	00B9	00BA	00BB	00BC	00BD	00BE	00BF	←Storage Positions

Note: The first operand does not include address 00B5.

Address Recall Register after Operation

00	B8

MOVE LOGICAL IMMEDIATE (MVI)

Operands	Op Code (hex)	Q-Byte[1] (binary)	Operand Address[2] (hex)	
	Byte 1	Byte 2	Byte 3	Byte 4
A1,I	3C	I	Operand 1 address	
D1(,R1),I	7C	I	Op 1 disp from XR1	
D1(,R2),I	BC	I	Op 1 disp from XR2	

[1] I = 1 byte of immediate data (for example, 1 byte of actual data on a 1-byte mask).

[2] Operand 1 is a 1-byte field; operand 2 is not used.

Operation

This machine instruction moves the Q-byte into operand 1.

Resulting Program Status Byte Settings

This machine instruction does not affect the program status register.

Example

Instruction

3C	AF	2F	CB

Operand 1 before Operation

00

2FCB ← Storage Position

Operand 1 after Operation

AF

2FCB ← Storage Position

SET BITS ON MASKED (SBN)

Operands	Op Code (hex)	Q-Byte[1] (binary)	Operand Address[2] (hex)	
	Byte 1	Byte 2	Byte 3	Byte 4
A1,I	3A	xxxx xxxx	Operand 1 address	
D1(,R1),I	7A	xxxx xxxx	Op 1 disp from XR1	
D1(,R2),I	BA	xxxx xxxx	Op 1 disp from XR2	

[1] The Q-byte contains a 1-byte binary mask specifying operand bits to be turned on.

[2] Operand 1 is a 1-byte field; operand 2 is not used.

Operation

The system looks at the Q-byte, bit by bit. If it finds a binary 1 in the Q-byte, it sets the corresponding bit in the operand byte to 1; if the system finds a binary 0 in the Q-byte, it does not change the corresponding bit in the operand.

Resulting Program Status Byte Settings

This machine instruction does not affect the program status register.

Example

Instruction

3A	01011010	00	20

Operand 1 before Operation

00001100

0020 ← Storage Position

Operand 1 after Operation

01011110

0020 ← Storage Position

SET BITS OFF MASKED (SBF)

Operands	Op Code (hex)	Q-Byte[1] (binary)	Operand Address[2] (hex)	
	Byte 1	Byte 2	Byte 3	Byte 4
A1,I	3B	xxxx xxxx	Operand 1 address	
D1(,R1),I	7B	xxxx xxxx	Op 1 disp from XR1	
D1(,R2),I	BB	xxxx xxxx	Op 1 disp from XR2	

[1] The Q-byte contains a 1-byte binary mask specifying operand bits to be turned on.

[2] Operand 1 is a 1-byte field; operand 2 is not used.

Operation

The system looks at the Q-byte, bit by bit. If it finds a binary 1 in the Q-byte, the system sets the corresponding bit in the operand byte to 0; if it finds a binary 0 in the Q-byte, it does not change the corresponding bit in the operand.

Resulting Program Status Byte Settings

This machine instruction does not affect the program status register.

Example

Instruction

3B	10000001	00	30

Operand 1 before Operation

01111001

0030 ← Storage Position

Operand 1 after Operation

01111000

0030 ← Storage Position

STORE REGISTER (ST)

Operands	Op Code (hex)	Q-Byte[1] (binary)	Operand Address[2] (hex)	
	Byte 1	Byte 2	Byte 3	Byte 4
A1,RX	34	Rx	Operand 1 address	
D1(,R1),RX	74	Rx	Op 1 disp from XR1	
D1(,R2),RX	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 machine 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 coding is:

Q-Byte Binary	Hex	Register Specified
0000 0000	00	None. The system ignores (no-ops) the instruction.
0000 0001	01	XR1.
0000 0010	02	XR2.
0000 0100	04	Program status register.
0000 1000	08	Address recall register.
0001 0000	10	Instruction address register.
0010 0000	20	Instruction address register.
0100 0000	40	Reserved; do not use.
1000 0000	80	Reserved; do not use.

Program Note

This machine instruction stores only one register at a time.

Resulting Program Status Byte Settings

This machine instruction does not affect the program status register.

Example

Instruction

34	00001000	32	BB

Address Recall Register

0A	CD

Operand before Operation

2F	C2

32BA 32BB ← Storage Positions

Operand after Operation

0A	CD

32BA 32BB ← Storage Positions

LOAD REGISTER (L)

Operands	Op Code (hex)	Q-Byte[1] (binary)	Operand Address[2] (hex)	
	Byte 1	Byte 2	Byte 3	Byte 4
A1,RX	35	Rx	Operand 1 address	
D1(,R1),RX	75	Rx	Op 1 disp from XR1	
D1(,R2),RX	B5	Rx	Op 1 disp from XR2	

[1] Rx specifies the register into which data is loaded.

[2] Operand 1 is a 2-byte field addressed by its rightmost byte; operand 2 is not used.

Operation

This machine instruction moves data from the 2-byte field specified by the operand address into the register specified by the Q-byte. The Q-byte coding is:

Q-Byte Binary	Hex	Register Specified
0000 0000	00	None. The system ignores (no-ops) the instruction.
0000 0001	01	XR1.
0000 0010	02	XR2.
0000 0100	04	Program status register.
0000 1000	08	Address recall register.
0001 0000	10	Instruction address register.
0010 0000	20	Instruction address register.
0100 0000	40	Reserved; do not use.
1000 0000	80	Reserved; do not use.

Program Notes

- This machine instruction loads only one register at a time.

- The six rightmost bits (bits 10-15) of the program status register are condition indicators. These bits are designated the program status byte throughout this manual. The other program status register bits are not used.

- You can use this machine instruction to perform an unconditional branch without changing the address recall register; load the branch to address into the instruction address register. At the end of this machine instruction, the program advances to the machine instruction at that address.

- If this instruction is used to load the program status register, the contents of the program status register will be as follows:

Q-byte bits	Resultant PSR bits
5 6 7	5 6 7
X 0 0	1 0 0
X 0 1	0 0 1
X 1 0	0 1 0
X 1 1	0 0 1

X can be either 1 or 0.

Resulting Program Status Byte Settings

This machine instruction does not affect the program status register unless that is the register specified by the Q-byte.

Example

Instruction

35	00000100	00	11

Operand

00000000	00000010

0010 0011 ←Storage Positions

Program Status Register before Operation

00000000	00110001

0 7 8 15 ←Bits

Byte 0 Byte 1

Program Status Register after Operation

00000000	00000010

0 7 8 15 ←Bits

Byte 0 Byte 1

LOAD INDEX REGISTER (LA)

Operands	Op Code (hex)	Q-Byte[1] (binary)	Operand Address[2] (hex)	
	Byte 1	Byte 2	Byte 3	Byte 4
A1,RX	C2	Rx	Direct address	
D1(,R1),RX	D2	Rx	Op 1 disp from XR1	
D1(,R2),RX	E2	Rx	Op 1 disp from XR2	

[1] Rx specifies the index register to be loaded:
XR1 = hex 01 or 03
XR2 = hex 02 or 00

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

Operation

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

Resulting Program Status Byte Settings

This machine instruction does not affect the program status register.

Example

Instruction

D2	02	05

Index Register 1

2A	15

Index Register 2 after Operation

2A	1A

TRANSFER (XFER)

Operand	Op Code (hex)	Q-Byte (hex)	R-Byte (hex)
	Byte 1	I2 Byte 2	I1 Byte 3
I1,I2	F5	xx	xx

Operation

If you use this instruction in your program, a program check will occur.

Resulting Program Status Byte Settings

This instruction does not affect the program status register.

Logical Machine Instructions

COMPARE LOGICAL CHARACTERS (CLC)

Operands	Op Code (hex)	Q-Byte[1] (hex)	Operand Addresses[2] (hex)			
	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
A1(L1),A2	0D	L1-1	Operand 1 address		Operand 2 address	
A1(L1),D2(,R1)	1D	L1-1	Operand 1 address		Op 2 disp from XR1	
A1(L1),D2(,R2)	2D	L1-1	Operand 1 address		Op 2 disp from XR2	
D1(L1,R1),A2	4D	L1-1	Op 1 disp from XR1	Operand 2 address		
D1(L1,R1),D2(,R1)	5D	L1-1	Op 1 disp from XR1	Op 2 disp from XR1		
D1(L1,R1),D2(,R2)	6D	L1-1	Op 1 disp from XR1	Op 2 disp from XR2		
D1(L1,R2),A2	8D	L1-1	Op 1 disp from XR2	Operand 2 address		
D1(L1,R2),D2(,R1)	9D	L1-1	Op 1 disp from XR2	Op 2 disp from XR1		
D1(L1,R2),D2(,R2)	AD	L1-1	Op 1 disp from XR2	Op 2 disp from XR2		

[1] The Q-byte designates the operand length:
L1-1 = the number of bytes in either operand, 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 machine instruction compares operand 1 with operand 2, byte by byte, and sets the program status register depending on the result of the compare. The compare looks at each operand as a binary quantity; that is, matching bytes from the two operands are compared, bit for bit.

Program Note

Neither operand is changed by the machine instruction.

Resulting Program Status Byte Settings

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 before and after Operation

27	FA	26	
0010	0011	0012	← Storage Positions

Operand 2 before and after Operation

23	FA	26	
0000	0001	0002	← Storage Positions

Program Status Byte before Operation

00100001	
0 7	← Bits

Program Status Byte after Operation

00100100	
0 7	← Bits

COMPARE LOGICAL IMMEDIATE (CLI)

Operands	Op Code (hex)	Q-Byte[1] (binary)	Operand Address[2] (hex)	
	Byte 1	Byte 2	Byte 3	Byte 4
A1,I	3D	I	Operand 1 address	
D1(,R1),I	7D	I	Op 1 disp from XR1	
D1(,R2),I	BD	I	Op 1 disp from XR2	

[1] I = 1 byte of immediate data (that is, 1 byte of actual data that is to be used in binary form).

[2] Operand 1 is a 1-byte field; operand 2 is not used.

Operation

This machine instruction compares all the bits in the Q-byte with all the bits in operand 1 and stores the result in the program status byte.

Program Note

Neither the Q-byte nor operand 1 is changed by this operation.

Resulting Program Status Byte Settings

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

Operand 1 before and after Operation

75

0021 ← Storage Position

Program Status Byte after Operation

00000010

0 7 ← Bits

TEST BITS ON MASKED (TBN)

Operands	Op Code (hex)	Q-Byte[1] (binary)	Operand Address[2] (hex)	
	Byte 1	Byte 2	Byte 3	Byte 4
A1,I	38	xxxx xxxx	Operand 1 address	
D1(,R1),I	78	xxxx xxxx	Op 1 disp from XR1	
D1(,R2),I	B8	xxxx xxxx	Op 1 disp from XR2	

[1] The Q-byte contains a 1-byte binary mask specifying operand bits for testing.

[2] Operand 1 is a 1-byte field; operand 2 is not used.

Operation

This machine instruction tests specified bits in the operand byte to see if they are on. For each mask bit (Q-byte bit) on, the system tests the matching bit in the operand. If any tested bit is off, the system turns the test false indicator (in the program status register) on.

Program Notes

- The operand and Q-byte are not changed.
- The test false condition is turned off by system reset, by using test false as a condition in a branch on condition or a jump on condition instruction, or by loading a binary 0 into program status register bit 11 (bit 3 of the rightmost program status register byte).

Resulting Program Status Byte Settings

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

Operand 1 before and after Operation

10010101

0021 ← Storage Position

Program Status Byte after Operation

00010000

0 7 ← Bits

TEST BITS OFF MASKED (TBF)

Operands	Op Code (hex)	Q-Byte[1] (binary)	Operand Address[2] (hex)	
	Byte 1	Byte 2	Byte 3	Byte 4
A1,I	39	xxxx xxxx	Operand 1 address	
D1(,R1),I	79	xxxx xxxx	Op 1 disp from XR1	
D1(,R2),I	B9	xxxx xxxx	Op 1 disp from XR2	

[1] The Q-byte contains a 1-byte binary mask specifying operand bits for testing.

[2] Operand 1 is a 1-byte field; operand 2 is not used.

Operation

This machine instruction tests specified bits in the operand byte to see if they are on. For each mask bit (Q-byte bit) that is a 1, the system tests the matching bit in the operand. If any tested bit is a 1, the system turns the test false indicator (in the program status register) on.

Program Notes

- The operand and Q-byte are not changed.
- The test false condition is turned off by system reset, by using test false as a condition in a branch on condition or jump on condition instruction, or by loading a binary 0 into program status register bit 11 (bit 3 of the rightmost program status register byte).

Resulting Program Status Byte Settings

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 on
2	Binary overflow	Bit not affected

Example

Instruction

39	01101100	00	25

Operand 1 before and after Operation

10010100

0025 ← Storage Position

Program Status Byte after Operation

00010000

0 7 ← Bits

BRANCH ON CONDITION (BC)

Operands	Op Code (hex)	Q-Byte[1] (binary)	Branch To Address (hex)	
	Byte 1	Byte 2	Byte 3	Byte 4
A1,I	C0	xxxx xxxx	Direct address	
D1(,R1),I	D0	xxxx xxxx	Disp from XR1	
D1(,R2),I	E0	xxxx xxxx	Disp from XR2	

[1] The Q-byte contains a binary mask specifying which program status register positions are tested by the instruction.

Operation

This machine instruction tests the program status register (rightmost byte) under control of the Q-byte. If the register meets the condition set up by the Q-byte, the system places the address of the next sequential machine instruction in the address recall register, places the branch to address in the instruction address register, and branches to the branch to address. If the register does not meet at least one condition set up by the Q-byte, the system places the address of the next sequential machine instruction in the instruction address register, and the program advances to the next sequential machine instruction.

The Q-byte determines what conditions are tested and if the branch is to occur on condition true (when the specified program status register bit is 1) or condition false (when the specified program status register bit is 0). When bit 0 of the Q-byte is 1, the branch occurs on condition true; when bit 0 is 0, the branch occurs on condition false.

Bits 2 through 7 of the Q-byte determine the bits to be tested in the program status register. These bits, and the conditions they represent, are:

Bit	Condition Tested
1	None (bit should be set to 0)
2	Binary overflow
3	Test false
4	Decimal overflow
5	High
6	Low
7	Equal

When bit 0 of the Q-byte is 1 (condition true), the branch occurs if any of the indicators tested is 1 (associated bit is 1). When bit 0 of the Q-byte is 0 (condition false), the branch occurs if all of the indicators tested are 0 (associated bits are all zero).

Program Notes

- The address placed in the address recall register remains there until a decimal add, decimal subtract, insert and test characters, zero and add zoned, load register, add to register, or another branch on condition machine instruction is executed.

- The program status byte is never equal to hex 00:
  - A Q-byte of hex 80, x7, or xF (where x is 0, 1, 2, 3, 4, 5, 6, or 7) causes the system to ignore the machine instruction (no operation occurs).
  - A Q-byte of hex 00, x7, or xF (where x is 8, 9, A, B, C, D, E, or F) causes an unconditional branch.

Resulting Program Status Byte Settings

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

C0	10001000	02	BF	
0BCC	0BCD	0BCE	0BCF	← Storage Pos

Program Status Byte before Operation

00011001

Instruction Address Register after Operation

02	BF

Address Recall Register after Operation

0B	D0

Program Status Byte after Operation

00010001	
0 7	← Bits

JUMP ON CONDITION (JC)

Operand	Op Code (hex)	Q-Byte[1] (hex)	R-Byte[2] (hex)
	Byte 1	Byte 2	Byte 3
A1,I	F2	xxxx xxxx	IAR disp

[1] The Q-byte contains a binary mask that indicates which status register bits (the bits in the rightmost byte of the program status register) are tested by the machine instruction.

[2] The R-byte is a displacement which, when added to the address in the machine instruction address register, provides a jump to address.

Operation

This machine instruction tests the rightmost byte of the program status register under control of the Q-byte. If the register meets the conditions set up by the Q-byte, the system adds the value stored in the instruction R-byte (byte 3) to the contents of the instruction address register and stores the result in the instruction address register. The program jumps to the new address stored in the instruction address register at the end of the jump on condition operation. If the register does not meet the condition(s) set up by the Q-byte, the system advances to the next sequential machine instruction in the program. The Q-byte determines what conditions are tested and if the jump is to occur on condition true (when the specified program status register bit is 1) or condition false (when the specified program status register bit is 0). When bit 0 of the Q-byte is 1, the jump occurs on condition true; when bit 0 of the Q-byte is 0, the jump occurs on condition false.

Bits 2 through 7 of the Q-byte determine the bits to be tested in program status register's rightmost byte. These bits, and the conditions they represent, are:

Bit	Condition Tested
1	None (bit should be set to 0)
2	Binary overflow
3	Test false
4	Decimal overflow
5	High
6	Low
7	Equal

When bit 0 of the Q-byte is 1 (condition true), the jump occurs if any of the indicators tested is on (associated bit is 1). When bit 0 of the Q-byte is 0 (condition false), the jump occurs if all of the indicators tested are off (associated bits all are 0).

Program Notes

- The program status byte is never equal to hex 00:
  - A Q-byte of hex 80, x7, or xF (where x is 0, 1, 2, 3, 4, 5, 6, or 7) causes the system to ignore the machine instruction (no operation occurs).
  - A Q-byte of hex 00, x7, or xF (where x is 8, 9, A, B, C, D, E, or F) causes an unconditional jump.

Resulting Program Status Byte Settings

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	←—— Storage Positions

Instruction Address Register after Operation

0B	CF

Program Status Byte before Operation

00001001	
0 7	←—— Bits

Program Status Byte after Operation

00001001	
0 7	←—— Bits

LOAD PROGRAM MODE REGISTER (LPMR)

Operand	Op Code (hex)	Q-Byte (hex)	R-Byte (hex)
	Byte 1	I2 Byte 2	I1 Byte 3
I1,I2	F6	[1]	I

[1] The Q-byte is ignored but must contain valid parity.

Operation

This instruction loads the program mode register with immediate data from the R-byte if program mode register bit 7 (nonprivileged mode bit) is not on. The contents of the program mode register bits 0 and 4 through 7 are replaced by the corresponding values in the R-byte. Bits 1 through 3 are ignored since these bits are not used by the register.

The Q-byte is ignored but must contain valid parity.

If this instruction is used when program mode register bit 7 is on (program is not privileged), the instruction execution is inhibited and a main storage processor storage exception check occurs.

Resulting Program Status Byte Settings

This instruction does not affect the program status register.

SUPERVISOR CALL (SVC)

Operand	Op Code (hex)	Q-Byte (hex)	R-Byte (hex)
	Byte 1	Byte 2	Byte 3
I2,I1	F4	xx	xx

The supervisor call instruction stops the main storage processor and generates an interrupt to the control processor. The supervisor call processor saves the current status of the main storage processor and uses the R-byte to determine what type of supervisor call has been requested. The individual supervisor call handlers use the Q-byte to further define the requested function.

The two primary types of supervisor calls are:

- Immediate supervisor call–The request is processed immediately by the specified supervisor call handler. Upon completion, control returns to the requesting task and the main storage processor is restarted.

- Delayed supervisor call–Data related to this request is saved in an action control element and chained to one of the delayed supervisor call processor handler's queues. Control is optionally returned to the requesting task (see the Q-bytes defined for each delayed supervisor call) or to another task ready to use the main storage processor.

The supervisor call instructions are 3 to 6 bytes long, depending upon the number of inline parameters passed with the request. The format and function of each supervisor call is described on the following pages. Control returns to the requesting task immediately following the last byte of the requested supervisor call instruction. A supervisor call op code cannot reside within the last 5 bytes of the last 2K pages of a task.

GENERAL WAIT

Op Code (hex)	Q-Byte (hex)	R-Byte (hex)	Inline Parameter List (hex)	
Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
F4	xx	00	xx	xx

The general wait supervisor call instruction allows multiple tasks to wait on a specified condition. There are 16 separate conditions which may be passed to the general wait handler through the inline parameter list (bytes 4 and 5). The conditions passed are saved in the caller's task control block in the field's TCBWMASK and TCBWMSK2. The task will not receive control of the main storage processor until one of the conditions being waited on is posted using supervisor call 01 (general post).

The first 8 bits of the inline parameter list (byte 4) are reserved for use by the control storage supervisor:

Bit	Meaning If On
0	System queue space (SQS) failure (supervisor call 06 or 0A)
1	Task work area allocate failure
2	Test and set failure (supervisor call 23)
3	Work station queue space (WSQS) failure (supervisor call 06 or 0A)
4	Resource enqueue failure (supervisor call 21)
5	Reserved
6	Sector enqueue failure (supervisor call 29)
7	Reserved

This supervisor call is privileged.

Input Parameter

Q-Byte: Bit	Meaning If On
0-5	Not used; must be zero.
6	Main storage processor transient area is not refreshable. The current transient is locked into the transient area for the duration of the wait. This bit is effective only when specified in instructions executed in the main storage transient area.
7	Not used; must be zero.

Inline Parameter 1: General wait mask (byte 4).

Inline Parameter 2: General wait mask (byte 5).

Example

Program A wants to be placed into a wait state on the printer. Another program (program B) currently owns the printer resource. When program B is done with the printer, the program always issues a general post (supervisor call 01). Program A issues the following supervisor call:

Assembler	Object Code
SVC 0,0 DC XL2'0080'	F400000080

Program A will wait until the general post before continuing.

In this example, program A and program B have adopted the convention of using 0080 for the printer.

GENERAL POST

Op Code (hex)	Q-Byte (hex)	R-Byte (hex)	Inline Parameter List (hex)	
Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
F4	00	01	xx	xx

The general post supervisor call instruction allows a task to post all tasks waiting on a specified condition. These tasks are made ready and will receive control based on their current priorities. The conditions specified on the general post supervisor call must match the conditions specified on the general wait (supervisor call 00). The first inline parameter list byte (byte 4) is reserved for use by the control storage supervisor.

A general post affects only those tasks in a general wait at the time the general post is issued. A general wait issued any time after the general post will not be satisfied by that general post. The general wait task will remain waiting.

This supervisor call is privileged.

Input Parameters

Q-Byte: Not used; must be zero.

Inline Parameter 1: General wait mask post code (byte 4)

Inline Parameter 2: General wait mask post code (byte 5)

Example

A program owns the printer resource and now wants to free it. A general post is issued for any program currently in a wait state on the printer resource.

```
Assembler        Object Code

SVC 1,0          F400010080
DC XLC'0080'
```

In this example, the use of 0080 is by convention of the main storage program issuing SVC 00 and SVC 01.

Note: If no tasks are currently waiting for the printer resource, this supervisor call will not post any tasks.

EVENT WAIT

Op Code (hex)	Q-Byte (hex)	R-Byte (hex)
Byte 1	Byte 2	Byte 3
F4	xx	02

The event wait supervisor call instruction allows the user to wait for completion of a specified event (specific wait) or any one of a group of events (multiple wait). For specific waits, the address of the event control mask being waited on must be in index register 1 (XR1). The event is defined as complete if the second byte of the event control mask has bit 1 = 1. For multiple waits, index register 1 must be set to 0000 before the event wait supervisor call is issued. In this case, the wait is satisfied if any event is complete for this task and bit 0 = 1 in the first byte of the event control mask. When a multiple wait is satisfied, control returns with index register 1 containing the address of the completed event control mask.

This supervisor call is privileged.

Input Parameters

Q-Byte: **Bit**	**Meaning If On**
0-4, 7	Not used; must be zero.
5	The address of the event to be waited on is a real address.
6	The transient area is not refreshable. The current transient is locked into the transient area for the duration of the wait. This takes effect only when issued from instructions executed in the main storage transient area.

Index Register 1: Contains the address of the event to be tested for completion. If index register 1 = 0000, this is a multiple wait.

Output Values

Index Register 1: Contains the address of the event that has satisfied this wait.

Example

A user issues a request for disk I/O. Index register 1 contains the address of the disk input/output block (hex 3000). The user issues an event wait supervisor call to wait for completion of the disk operation.

Assembler	**Object Code**
SVC 2,0	F40002

with index register 1 containing address 3000; locations 3000 and 3001 contain 0000.

When the disk operation is completed without error, control returns with locations 3000 and 3001 (the event control mask) containing hex 0040.

Note: Address 3000 is a translated address.

EVENT POST

Op Code (hex)	Q-Byte (hex)	R-Byte (hex)	Inline Parameter List (hex)	
Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
F4	xx	03	xx	xx

This event post supervisor call instruction indicates that a specified event is complete. The event post instruction uses the 2-byte event control mask whose address is given by the requesting task in index register 1 (XR1). The second byte of the event control mask is altered to hex 4n where n is the completion code given by the user in the second inline parameter byte (byte 5).

The first inline parameter byte (byte 4) specifies the device type associated with the event. An action control element, related to the event, must have previously been queued to the system queue associated with the specified device type. If the action control element is not found, the event post request is ignored.

If the task associated with the event being posted is waiting for the event to complete, the task will be made ready and will be given control of the main storage processor based on its current priority. The caller of event post may set on the *do not preempt* bit in the second inline parameter byte (byte 5) to prevent the posted task from gaining immediate control.

This supervisor call is privileged.

Note: The event address must be in real storage.

Input Parameters

Q-Byte: **Bit**	**Meaning If On**
0-5, 7	Not used; must be zero.
6	Transient area is not refreshable. The current transient is locked into the transient area for the duration of the wait. This takes effect only when issued from instructions executed in the main storage transient area.

Inline Parameter 1: Offset into system queues of action control element chain for this event.

Inline Parameter 2:

Bit	Meaning If On
0	Do not preempt task issuing event post supervisor call.
1	Last-in-first-out (LIFO) queue action control element to complete queue of task being posted.
2-3	Not used; must be zero.
4-7	Completion code (0-F).

Output Parameters

The TCB field TCB@INL4 contains the type byte (ACETYPE) of the action control element that was posted.

Example

Program A issues supervisor call 02 (SVC 02,00) with index register 1 equal to 1000. Program B then issues:

Assembler	Object Code
SVC 03,00 DC XL1'48' DC XL1'84'	F400034884

Index register 1 points to location 1000 and locations 1000 and 1001 contain 0000.

After event post, locations 1000 and 1001 contain 0044 and program A will be made ready to run in the main storage processor. Program A will not gain control until program B issues a wait since the *do not preempt* indicator is on.

Note: An action control element must be queued to the header at location 0148 when the event post supervisor call is issued.

TRANSFER CONTROL/SYSTEM TRANSIENT

Op Code (hex)	Q-Byte (hex)	R-Byte (hex)	Inline Parameter List (hex)
Byte 1	Byte 2	Byte 3	Byte 4
F4	xx	04	xx

The transfer control/system transient supervisor call allows programs to pass control to specified, privileged SSP routines.

The passed inline parameter is an offset into the resident transient/transfer control table. This table contains the address of the transient/transfer control routine to be called.

Transient Table Entry

Bytes 0, 1, 2	Disk sequential sector address of transient.
Byte 3	Length of transient (0-8 = actual length in sectors).

Transfer Control Table Entry

Bytes 0, 1	Main storage address of routine.
Byte 2	The program mode register setting to be used for this routine.
Byte 3	Bit 2 = 1 indicates that this routine is privileged. Bit 3 = 1 indicates that this is a transfer control table entry.

For transient requests, the caller's task control block is placed on the system queue for transients. The transient is given control when the transient area becomes available. Transfer routines are given control immediately.

This supervisor call is privileged for certain routines.

Input Parameters

Q-Byte: Bit	Meaning If On
0-5	Not used; must be zero.
6	The transient area is not refreshable. The current transient is locked into the transient area for the duration of the wait. This takes effect only when issued from instructions executed in the main storage transient area.
7	Return control to caller if caller is a transient or transfer control routine.

Inline Parameter 1: Offset in table of transfer/transient table entry.

Example

Program A issues a transfer control with inline parameter 1 equal to hex 13.

Assembler	Object Code
SVC 4,0 DC XL1'13'	F4000413

The value in the table related to hex 13 is 10808010.

After this supervisor call, program A is given control with the instruction address register equal to 1080 and dispatching disabled.

Note: Instruction address register translate and nonprivileged mode are always set off with a transfer control instruction.

FREE CURRENT REQUEST BLOCK

Op Code (hex)	Q-Byte (hex)	R-Byte (hex)
Byte 1	Byte 2	Byte 3
F4	00	05

The current request block is dequeued from the task control block request block chain and freed.

This supervisor call is privileged.

Input Parameters

Q-Byte: Not used; must be zero.

Output Values

The task control block field TCBCRB now contains the address of the second request block on the chain (or all zeros if only one request block was on the chain).

Example

Program A issues a free current request block supervisor call with TCBCRB = 0600 and locations 0600 and 0601 contain 0630.

After the user issues:

Assembler	Object Code
SVC 05,00	F40005

the TCBCRB = 0630.

ASSIGN

Op Code (hex)	Q-Byte (hex)	R-Byte (hex)	Inline Parameter List (hex)		
Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
F4	xx	06	xx	xx	xx

The assign supervisor call instruction allocates storage out of one of the system free space areas. Storage is assigned in multiples of 8 bytes on 8-byte boundaries. A set of eight system queue headers are reserved for various types of assigns. The area to use is indicated by the third inline parameter (byte 6).

An option to queue assigned areas to the requester's task control block is available using the third inline parameter (bit 0 = 1). The requester must add 4 bytes to the length of the area needed. The last 4 bytes of the area assigned are used by the control storage supervisor to maintain the queue. The requester must not alter this area.

If sufficient space is not available, the user may either wait for free space to become available or have control returned immediately with index register 1 set to all zeros (0000). Requests for large amounts of free space are more likely to fail since the area assigned is always contiguous space.

This supervisor call is privileged.

Input Parameters

Q-Byte	**Bit**	**Meaning If On**
	0-5	Not used; must be zero.
	6	Transient area is not refreshable. The current transient is locked into the transient area for the duration of the wait. This takes effect only when issued from instructions executed in the main storage transient area.
	7	Wait for assign area to become available.

Inline Parameters 1 and 2: Length of area to assign.

Inline Parameter 3: Type of request.

Bit	Meaning
0	(If = 1) Queue area to task control block and save length.
1-7	(If = 0000000) Use work station queue space.
1-7	(If = 0000001) Use system queue space.

Output Values

Index Register 1: Address of the area assigned (zero if no space is assigned and the no wait option of the Q-byte is specified).

Example

Program A issues the assign supervisor call to allocate 256 bytes of storage from system queue space (SQS). Program A is willing to wait for the storage if space is not immediately available.

Assembler	Object Code
SVC 06,01 DC XL2'0104' DC XL1'81'	F40106010481

After this supervisor call has executed, an area of hex 0108 is assigned to program A. The last 4 bytes are used to queue this area to program A's task control block. The address of the hex 0100 bytes allocated to the program is returned in index register 1 (that is, for example, index register 1 contains 4858).

Note: This is always an 8-byte boundary.

FREE ASSIGNED AREAS

Op Code (hex)	Q-Byte (hex)	R-Byte (hex)	Inline Parameter List (hex)		
Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
F4	xx	07	xx	xx	xx

The free instruction returns previously assigned storage to the system free space areas. This area freed is either merged to one of the current free areas (if adjacent) or queued to the free chain. The free area to use as well as the type of assign used when getting storage (queued or not queued) must be indicated. This supervisor call also permits a partial free of the originally assigned area (nonqueued only) as long as the area freed is a multiple of 8 bytes on an 8-byte boundary.

This supervisor call is privileged.

Input Parameters

Q-Byte: Not used; must be zero.

Inline Parameters 1 and 2: Length of area to free. (A value equal to the value assigned should be specified. The area freed will be a multiple of 8 bytes.)

Inline Parameter 3: Type of request.

Bit	Meaning
0	(If = 1) The area has been queued by assign.
1-7	(If = 0000000) Use work station queue space.
1-7	(If = 0000001) Use system queue space.

Index Register 1: Address of area to free.

Example

Program A wants to free 256 bytes of storage previously assigned from system queue space. The address of the area assigned is hex 4858. The original request was a queued request. Program A loads index register 1 with hex 4858 and issues:

Assembler	Object Code
SVC 07,00 DC XL2'0104' DC XL1'81'	F40007010481

The area is first dequeued from program A's task control block. Then the total area of hex 0108 bytes is freed and made available for other users of system queue space.

SENSE ADDRESS/DATA SWITCHES

Op Code (hex)	Q-Byte (hex)	R-Byte (hex)
Byte 1	Byte 2	Byte 3
F4	xx	09

This function allows a main storage program to sense the Address/Data switches. The value in these switches is returned in index register 1.

This supervisor call is nonprivileged.

Input Parameters

Q-Byte: Not used; must be zero.

Example

Assembler	Object Code
SVC 09,00	F40009

Control returns with index register 1 set to the current value of the Address/Data switches. (If the Address/Data switches are set to FF00, index register 1 will contain FF00.)

ASSIGN SYSTEM QUEUE SPACE

Op Code (hex)	Q-Byte (hex)	R-Byte (hex)	Inline Parameter List (hex)
Byte 1	Byte 2	Byte 3	Byte 4
F4	xx	0A	xx

The assign system queue space supervisor call instruction allocates storage out of one of the system free space areas. Storage is assigned in multiples of 8 bytes on 8-byte boundaries. A set of eight system queue headers are reserved for various types of assigns. The area to use is indicated by inline parameter 1 (byte 4).

An option to queue assigned areas to the requester's task control block is available using inline parameter 1 (byte 4–bit 0 = 1). The requester must add 4 bytes to the length of the area needed. The last 4 bytes of the area assigned are used by the control storage supervisor to maintain the queue. The requester must not alter this area.

If sufficient space is not available, the user may either wait for free space to become available or have control returned immediately with index register 1 set to all zeros (0000). Requests for large amounts of free space are more likely to fail since the area assigned is always contiguous space.

This supervisor call is privileged.

Input Parameters

Q-Byte: Bit	Meaning If On
0-5	Not used; must be zero.
6	Transient area is not refreshable. The current transient is locked into the transient area for the duration of the wait. This takes effect only when issued from instructions executed in the main storage transient area.
7	Wait for assign area to become available.

Inline Parameter 1: Type of request.

Bit	Meaning
0	Queue area to task control block and save length.
1-7	(Equal to 0000000) Use work station queue space.
1-7	(Equal to 0000001) Use system queue space.

Index Register 1: Length of area to assign.

Output Values

Index Register 1: Address of the area assigned (zero if no space is assigned and the no wait option of the Q-byte is specified.)

Example

Program A wants to assign N bytes of storage from work station queue space (WSQS). The length of this area is passed to program A. Program A will not wait for the storage if it is not available.

Assembler	Object Code
L LENGTH (,XR2),XR1	
SVC 10,00	F4000A00
DC XL1'00'	

After the execution of this supervisor call, program A's index register 1 contains the address of the area assigned (3290) or all zeros (0000) if the length requested was not available.

POST ACTION CONTROLLER STATUS WORD

Op Code (hex)	Q-Byte (hex)	R-Byte (hex)	Inline Parameter List (hex)
Byte 1	Byte 2	Byte 3	Byte 4
F4	00	0B	xx

This routine posts the appropriate control storage routine to be executed. The control storage routine will receive control based on its priority after the supervisor call has executed.

This supervisor call is privileged.

Input Parameters

Q-Byte: Not used; must be zero.

Inline Parameter 1: The mask value associated with the action controller routine to be posted.

Example

Program A issues the following instruction:

Assembler	Object Code
SVC 0,11 DC XL1'17'	F4000B17

The action controller routine associated with the value hex 17 will be given control after this routine is executed. The exact time control is passed to the routine is dependent upon other work being done by the control storage processor.

LOAD ADDRESS TRANSLATION REGISTERS

Op Code (hex)	Q-Byte (hex)	R-Byte (hex)
Byte 1	Byte 2	Byte 3
F4	00	0C

This instruction loads the task address translation registers. The values loaded into the address translation registers are the values found in the caller's task control block (TCB) field TCBATRS.

This supervisor call is privileged.

Input Parameters

Q-Byte: Not used; must be zero.

Example

Program A issues the following instruction:

Assembler	Object Code
SVC 0,12	F4000C

with the TCBATR field set to:

0C0D0E0F101112FFFFFFFFFFFFFFFFFF

FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

After executing the load address translation registers instruction, the values above are loaded into the task address translation registers. These values will be used for all translated requests.

SET PROGRAM MODE REGISTER

Op Code (hex)	Q-Byte (hex)	R-Byte (hex)	Inline Parameter List (hex)	
Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
F4	00	0D	xx	xx

This instruction alters the main storage caller's program mode register according to the inline parameter list.

This supervisor call is privileged.

Input Parameters

Q-Byte: Not used; must be zero.

Inline Parameter 1:

Byte 1 (hex)

80	Disable task dispatching
70	Reserved
08	Turn on instruction address register translation[1]
04	Turn on operand 2 address translation
02	Turn on operand 1 address translation
01	Change from privileged to nonprivileged mode

Byte 2 (hex)

80	Enable task dispatching
70	Reserved
08	Turn off instruction address register translation[1]
04	Turn off operand 2 address translation
02	Turn off operand 1 address translation

Example

Program A issues the following instruction:

Assembler	Object Code
SVC 0,13 DC XL2'8400'	F4000D8400

with the program mode register set to hex 0A.

After the instruction is executed, the program mode register contains hex 8E.

[1] Altering the instruction address register translation results in branching from the current value in the instruction address register, under current translation, to the same address in the opposite translation.

QUEUE/DEQUEUE

Op Code (hex)	Q-Byte (hex)	R-Byte (hex)	Inline Parameter List (hex)		
Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
F4	00	0E	xx	xx	xx

This instruction provides the functions to maintain the system queues. The functions provided are:

- Queue FIFO (first-in-first-out): Place block at end of queue.

- Queue LIFO (last-in-first-out): Place block first on queue.

- Queue FIFO by priority: Place block at end of blocks of equal priority.

- Queue LIFO by priority: Place block at front of blocks of equal priority.

- Dequeue.

These functions are provided for any of the system queues or for a queue header passed by the caller. The chaining field may be located anywhere within the first 256 bytes of the block to be manipulated. The priority field, if given, must be located within the first 16 bytes of the block to be queued.

No check is made for an invalid queue (that is, never ending or recursive queue). All blocks must be located in real storage.

This supervisor call is privileged.

Input Parameters

Q-Byte: Not used; must be zero.

Inline Parameter 1: Displacement of queue header from start of system queue headers (for system requests)–left byte.

Inline Parameter 2: Displacement into control block of chaining field (range 0 to 255)–left byte.

Inline Parameter 3: Queuing indicators and priority field displacement.

Bit	Meaning If On
0	Priority request
1	System request (system queue header passed)
2	Dequeue request
3	LIFO request
4-7	Priority field displacement (0-F)

Index Register 1: Address of control block.

Index Register 2: Address of queue header for nonsystem requests.

Example

Program A wants to FIFO queue control block X to the system queue located at offset hex 4E into the system queues. The offset into the control block of the chaining field is hex 06.

Assembler	Object Code
SVC 0,14 DC XL3'4E0640'	F4000E4E0640

with index register 1 containing 34A8, location 34A8 and the following locations containing 000044008700329000.

Currently the given system queue has the following values:

014E = 42F0
42F0 = 000000000000338000
3380 = 000000000000000000

After the queue request:

014E = 42F0
42F0 = 000000000000338000
3380 = 00000000000034A800
34A8 = 000044008700000000

SYSTEM CONTROL BLOCK ACCESS

Op Code (hex)	Q-Byte (hex)	R-Byte (hex)	Inline Parameter List (hex)	
Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
F4	00	0F	xx	xx

This instruction allows a main storage user to access 2 bytes from a control storage direct area. This routine also allows a main storage user to pick up 2 bytes from a queue header in main storage.

This supervisor call is privileged.

Input Parameters

Q-Byte: Not used, must be zero.

Inline Parameter 1: Area and function.

Bit	Meaning

0-3	Identifies direct area to be used:
	If = 0000, direct area 0
	If = 0001, direct area 1
	If = 0010, direct area 2
	If = 0011, direct area 3

4-5	Not used; must be zero.
6	(If = 1) Queue header request.
7	(If = 1) Put request.

Inline Parameter 2: Displacement.

Index Register 2: Two-byte data area (if put request).

Output Parameters

Index Register 2: Data area retrieved (if get request).

Example

Program A wants to get 2 bytes from control storage location 2047.

Assembler	Object Code
SVC 0,15 DC XL2'2047'	F4000F2047

with location 2047 containing 0A42.

After executing the instruction, index register 2 contains 0A42.

MAIN STORAGE TRANSIENT SCHEDULER

Op Code (hex)	Q-Byte (hex)	R-Byte (hex)	Inline Parameter List (hex)	
Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
F4	xx	10	xx	xx

This instruction provides the same function as the transfer control/system transient instruction (R byte of 04). The address of the table entry is passed directly by the caller in inline parameters 1 and 2 (bytes 4 and 5).

This supervisor call is privileged.

Input Parameters

Q-Byte: **Bit** **Meaning If On**

0-5 Not used; must be zero.

6 The transient area is not refreshable. The current transient is locked into the transient area for the duration of the wait. This takes effect only when issued from instructions executed in the main storage transient area.

7 Return control to the caller if caller is a transient or transfer control routine.

Inline Parameters 1 and 2: Address in main storage of a transient/transfer control table entry (may be in either real or translated storage; however, it must be the same translation as the caller's instruction address register).

Example

Program A wants to load a transient from disk with the disk sequential sector address of the transient specified within the task storage.

Assembler **Object Code**

```
SVC 0,16       F400100842
DC  AL2(TRAN@)
```

where TRAN@ is at location 0842 and location 0842 contains 000F9206.

As a result of this supervisor call, the transient at the disk sequential sector address of 0F92 for a length of 6 sectors is loaded into the main storage transient area at location 0800. Control is given to the transient after it is loaded.

MAIN STORAGE TRANSIENT EXIT

Op Code (hex)	Q-Byte (hex)	R-Byte (hex)
Byte 1	Byte 2	Byte 3
F4	00	11

This instruction returns control to the caller of a transient/transfer routine. Control is returned at the next sequential instruction following the supervisor call (either 04 or 10). If the caller is not a transient, the transient area is set not busy. This allows other tasks to obtain the transient area. If the caller is a transient, the main storage transient exit instruction causes that transient to be reloaded into the transient area before passing control to the next sequential instruction.

Note: The nonrefresh capability described under supervisor call 04 may cause control to be returned to a module other than the immediate caller. If the nonrefresh indicator is used on supervisor call 04 or 10, control is returned to the most recent routine that did not specify nonrefresh.

This supervisor call is privileged.

Input Parameters

Q-Byte: Not used; must be zero.

Example

Transient X has completed execution and wants to return control to its caller. The main storage transient exit instruction is issued:

Assembler	Object Code
SVC 0,17	F40011

Control is returned to the next sequential instruction of the caller of transient X.

GET PAGE

Op Code (hex)	Q-Byte (hex)	R-Byte (hex)	Inline Parameter List (hex)		
Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
F4	xx	12	xx	xx	xx

This instruction expands the user's main storage size up to the region size. If enough main storage is not available to satisfy the request, the task is swapped out and in to acquire the storage. If the region requested cannot be obtained using all of main storage, the size available is given to the requester.

This supervisor call is nonprivileged.

Input Parameters

Q-Byte: **Bit**	**Meaning If On**
0-5,7	Not used; must be zero.
6	Transient area is not refreshable. The current transient is locked into the transient area for the duration of the wait. This takes effect only when issued from instructions executed in the main storage transient area.

Inline Parameters 1 and 2: Address of where to store last logical address plus 1.

Inline Parameter 3: Zero.

Output Parameters

The last logical address plus 1 is returned in the address passed by the caller in inline parameters 1 and 2.

Example

Program A is currently executing in 14K of main storage. When program A was requested, the user specified a region size of 24K. To acquire the additional storage, program A issues:

Assembler	**Object Code**
SVC 18,0	F40012220000
DC XL2'2200'	
DC XL1'0'	

After this supervisor call executes, location hex 2200 contains the address plus 1 of the last byte in program A's region. If program A originates at location 0000, and the full 24K was allocated, locations 2200 and 2201 will contain the value hex 6000.

Note: If only 20K was allocated to program A, then locations 2200 and 2201 will contain hex 5000.

FREE PAGE

Op Code (hex)	Q-Byte (hex)	R-Byte (hex)
Byte 1	Byte 2	Byte 3
F4	00	13

This instruction frees all the main storage pages owned by the requesting task.

This supervisor call is privileged.

Note: This instruction must be issued from real storage.

Input Parameters

Q-Byte: Not used; must be zero.

Example

Program A wants to free all main storage currently allocated:

Assembler	Object Code
SVC 0,19	F40013

Control is returned with all storage freed and the TCBATR field set to all hex FFs from (TCBBEGL) to (TCBMSSIZ).

If the TCBATR field contained:

```
000102030405060708090A0B0C0D0E0F
FFFFFFFFFFFFFFFFFF0B0C0D0E0F090A
```

TCBBEGL contained hex 19
TCBMSSIZ contained hex 07

then after the free page supervisor call, the TCBATR will contain:

```
000102030405060708090A0B0C0D0E0F
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
```

ASYNCHRONOUS TASK WAIT

Op Code (hex)	Q-Byte (hex)	R-Byte (hex)	Inline Parameter List (hex)
Byte 1	Byte 2	Byte 3	Byte 4
F4	00	17	xx

This instruction places the task control block (TCB) specified in index register 1 into a wait state. The type of wait (defined under supervisor call 1E, task wait) is specified using inline parameter 1.

This supervisor call is privileged.

Input Parameters

Q-Byte: Not used; must be zero.

Inline Parameter 1: Wait type.

Index Register 1: Address of task control block to be placed in a wait state.

Example

Program A and program B are executing under two different TCBs. Program A wants to suspend the execution of program B.

```
Assembler          Object Code

L @BTCB,XR1
SVC 23,0           F4001702
DC XL1'02'
```

Program B's TCB will be placed in a suspend wait until it is posted by program A using supervisor call 1D.

SET TRANSIENT AREA NOT BUSY

Op Code (hex)	Q-Byte (hex)	R-Byte (hex)
Byte 1	Byte 2	Byte 3
F4	00	18

The main storage transient area is set not busy if the caller is the transient area owner. The main storage transient scheduler will now schedule the next task waiting for the transient area.

This supervisor call is privileged.

Input Parameters

Q-Byte: Not used, must be zero.

Example

Program A issued a fetch request from the transient area (supervisor call 52). The module loaded receives control outside the transient area. This module needs to release the transient area for other programs:

Assembler	Object Code
SVC 24,0	F40018

POST ACTION CONTROL ELEMENT

Op Code (hex)	Q-Byte (hex)	R-Byte (hex)	Inline Parameter List (hex)	
Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
F4	00	19	xx	xx

This instruction posts an event complete using the action control element as input. The event address is retrieved from the action control element (in the index register 1 save area field) and the event is posted as in event post (supervisor call 03).

This supervisor call is privileged.

Input Parameters

Q-Byte: Not used; must be zero.

Inline Parameter 1: The queue header displacement where the action control element can be found.

Inline Parameter 2: Completion code.

Bit	Meaning
0	(If = 1) Do not preempt.
1	(If = 1) Queue last-in-first-out to task control block complete queue.
2-3	Not used.
4-7	Completion code (0-F).

Index Register 1: Action control element address.

Output Parameters

The TCB field TCB@INL4 contains the type byte (ACETYPE) of the action control element that was posted.

Example

Program A wants to post an event complete. The event was created using supervisor call 4C. The address of the action control element created by supervisor call 4C is hex 3478. Program A issues:

Assembler	Object Code
SVC 25,0 DC XL1'48' DC XL1'00'	F400194800

After completion of this supervisor call, the action control element at location hex 3478 will be dequeued from the system header at locations 0148 and 0149 and queued to the task control block field TCBCMPLQ of the task that issued the supervisor call 4C.

LOG TRACE INFORMATION

Op Code (hex)	Q-Byte (hex)	R-Byte (hex)
Byte 1	Byte 2	Byte 3
F4	00	1A

This instruction logs trace entries in the resident main storage trace buffer. The 15 bytes of data addressed by index register 2 are moved into the trace buffer. The trace data must be translated the same as the current instruction address register translation.

This supervisor call is nonprivileged.

Input Parameters

Q-Byte: Not used; must be zero.

Index Register 2: Address of information (15 bytes) to be placed in trace buffer.

Example

Program A wants to log 15 bytes of data to the system trace table. The data is at locations hex 2000 to 200E.

```
Assembler        Object Code

LA TRACE, XR2    C2022000
SVC 26,0         F4001A
```

SCAN SYSTEM QUEUE

Op Code (hex)	Q-Byte (hex)	R-Byte (hex)	Inline Parameter List (hex)	
Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
F4	00	1B	xx	xx

This instruction allows you to scan a queue for a specific value in one of the blocks on the queue. The search argument is passed by the caller and the address of the block containing the value is returned.

This supervisor call is privileged.

Input Parameters

Q-Byte: Not used; must be zero.

Inline Parameter 1: Argument displacement (must be 2-byte field)–left byte.

Inline Parameter 2: Chain field displacement–left byte.

Index Register 1: Search argument (2 bytes).

Index Register 2: Queue header (address of queue to be scanned).

Output Parameters

Index Register 2: Contains the address of the control block containing the passed argument. If the argument is not found, index register 2 is set to 0000.

Example

Program A wants to scan the disk spindle A1 active queue to find if the task control block for this program has any request pending.

```
Assembler        Object Code

L @ATCB,XR1
L X'0130',XR2
SVC 27,0         F4001B0E00
DC AL1(ACETCB - 1)
DC AL1(ACECHAIN - 1)
```

If an action control element (ACE) exists on the specified queue with the task control block address specified in index register 1, the address of the action control element is returned in index register 2. Otherwise, index register 2 is returned set to zeros.

TASK POST

Op Code (hex)	Q-Byte (hex)	R-Byte (hex)	Inline Parameter List (hex)
Byte 1	Byte 2	Byte 3	Byte 4
F4	00	1D	xx

The task post supervisor call posts a task's execution status. Inline parameter 1 contains the condition to be posted. TCBSTAT2 contains the task wait conditions that are turned off when the task post supervisor call is requested. If TCBSTAT2 becomes all zeros as a result of the post, the task is ready for execution. The wait conditions available for posting from main storage are:

- Task suspended
- Dedicated task wait

All other wait codes are handled internally by the control storage supervisor and should be posted only when purging associated requests.

This supervisor call is privileged.

Input Parameters

Q-Byte: Not used; must be zero.

Inline Parameter 1: Task post conditions.

Index Register 1: Task control block address of task to be posted.

Example

Program A has been suspended by an earlier task wait. To restart program A, the following supervisor call is issued:

```
Assembler      Object Code

L  @TASKA,1
SVC 29,0       F4001D02
DC  XL1'02'
```

TASK WAIT

Op Code (hex)	Q-Byte ·(hex)	R-Byte (hex)	Inline Parameter List (hex)
Byte 1	Byte 2	Byte 3	Byte 4
F4	xx	1E	xx

This instruction places the requesting task into a wait state. Inline parameter 1 is read into TCBSTAT2. The task remains in the wait state until all the bits in TCBSTAT2 are set off by task post (supervisor call 1D). In addition, if task dispatching is disabled, task wait automatically continues for the duration of the wait. The valid conditions to be set on in TCBSTAT2 are:

- Event wait
- Transient area wait
- General wait
- Internal control storage wait
- Timer wait
- Dedicated program wait
- Suspension wait

This supervisor call is privileged.

Input Parameters

Q-Byte: **Bit**	**Meaning If On**
0-5,7	Not used; must be zero.
6	Transient area is not refreshable. The current transient is locked into the transient area for the duration of the wait. This takes effect only when issued from instructions executed in the main storage transient area.

Inline Parameter 1: Wait conditions to be set on in TCBSTAT2.

Example

Program A wants to place itself into a dedicated wait. Program A issues:

Assembler	**Object Code**
SVC 30,0	F4001E08
DC XL1'08'	

Program A will wait until its task control block is specifically posted by another program for the dedicated wait condition.

RESOURCE ENQUEUE/DEQUEUE

Op Code (hex)	Q-Byte (hex)	R-Byte (hex)	Inline Parameter List (hex)
Byte 1	Byte 2	Byte 3	Byte 4
F4	xx	21	xx

This instruction enqueues and dequeues allocation queue elements (AQE) on a queue. It checks the share level of each queued entry and processes each accordingly.

For enqueues:

If the caller can share with the current owner of the queue (or if no owner exists), control is returned to the caller with an allocation queue element queued to the passed queue and the equal condition set in the program status register. If the caller's task cannot share with the owning task, a nonequal program status condition is returned (low if the owner is never ending or his task is suspended, high otherwise).

Optionally, a caller may wait for the resource to become available (Q code, bit 7 = 1). In this case, control will always return with equal program status condition.

For dequeues:

The allocation queue element allocated for this task is removed from the passed queue. If an allocation queue element for the caller is not queued, a nonequal program status register is returned.

This supervisor call is privileged.

Input Parameters

Q-Byte: Bit	Meaning If On
0-5	Not used; must be zero.
6	Transient area is not refreshable. The current transient is locked into the transient area for the duration of the wait. This takes effect only when issued from instructions executed in the main storage transient area.
7	Wait.

Inline Parameter 1:

Bit 0 = 1	Enqueue request
Bit 1 = 1	NEP requestor
Bit 2, 3	Not used
Bit 4-7	0000 Level 0 (shares with 0, 1, 2) 0001 Level 1 (shares with 0, 1) 0010 Level 2 (shares with 0) 0011 Level 3 (does not share)

Index Register 2: Address of queue header (left byte where allocation queue element is to be built).

Example

Program A and program B want to use the same common area. To prevent interferring with each other, both programs agree to enqueue on the resource by defining a queue header at locations hex 0190 and 0191 in main storage. This queue header represents the resource for programs A and B. When program A issues:

```
Assembler          Object Code

LA  X'0190',XR2    C2020190
SVC 33,1           F4012103
DC  XL1'03'
```

Program A will be allocated the resource by an allocation queue element (AQE) queued to locations hex 0190 and 0191. If program B now tries to allocate the same resource, program B's task control block will be placed in a general wait (if the Q-byte bit 7 = 1). When program A has completed its access of the common area, program A issues:

```
Assembler          Object Code

LA  X'0190',XR2    C2020190
SVC 33,0           F4002183
DC  XL1'83'
```

The allocation queue element that was queued to locations hex 0190 and 0191 is freed and dequeued. Program B is automatically given control of the resource and an allocation queue element is created to indicate program B's task control block is now the owner of the specified resource.

Note: This example shows two program requesting exclusive ownership of a resource. Various levels of sharing are possible using levels 0, 1, and 2.

DUMP MAIN STORAGE/TERMINATE TASK

Op Code (hex)	Q-Byte (hex)	R-Byte (hex)	Inline Parameter List (hex)	
Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
F4	00	22	xx	xx

This instruction terminates the calling task with a message identification (error) code 0016. Storage is dumped to disk and the instruction address register for the issuing task is set to 0092 (the end of job supervisor call) in the system communications area. The abnormal termination bit in the task control block is set on. The message identification code (MIC) is stored in the task control block.

This supervisor call is nonprivileged.

Input Parameters

Q-Byte: Not used; must be zero.

Inline Parameters 1 and 2: Hexadecimal value of the MIC number to be used to indicate the error. A value less than hex 0100 will cause the system to proc check. Values above hex 00FF will cause messages to be issued with options as defined for MIC 0001.

Example

Program A wants to terminate with a dump. Program A issues:

Assembler	Object Code
SVC 34,0	F40022

All of control storage and main storage are dumped to the system dump file and the program set for abnormal termination.

TEST AND SET

Op Code (hex)	Q-Byte (hex)	R-Byte (hex)	Inline Parameter List (hex)
Byte 1	Byte 2	Byte 3	Byte 4
F4	xx	23	xx

This instruction allows a main storage routine to test a selected bit in storage and set the bit on at the same time. If the bit is already on, the task is put into a general wait.

This supervisor call is privileged.

Input Parameters

Q-Byte:

Bit	Meaning If On
0-4	Not used; must be zero.
5	Byte is in real storage.
6	Transient area is not refreshable. The current transient is locked into the transient area for the duration of the wait. This takes effect only when issued from instructions executed in the main storage transient area.
7	Do not wait if bit is already on.

Inline Parameter 1: Value is bit to be tested.

Index Register 1: Address (minus one) of the byte to be tested and set.

Output Values

Program Status Register: False if the bit is on and the no wait Q-byte is specified.

Example

Program A and program B want to exclude one another when accessing a common area. The common area is located at 005C in main storage. Both programs issue the following supervisor call before accessing the common area:

```
Assembler           Object Code

LA X'005B',XR1
SVC 35,4            F4042302
DC XL1'02'
```

If bit 6 at location hex 005C is on when the supervisor call is issued by program A, then program A is placed in a general wait. If the bit is off, it is set on and control returned immediately to program A.

After accessing the common area, both program A and program B must set the bit off and issue a supervisor call 01 for the test and set failure condition.

TASK CONTROL BLOCK PRIORITY QUEUE

Op Code (hex)	Q-Byte (hex)	R-Byte (hex)	Inline Parameter List (hex)
Byte 1	Byte 2	Byte 3	Byte 4
F4	00	24	xx

This instruction stores the new task priority and passes to it into the task control block. The instruction then dequeues the task control block from the task priority queue and requeues it by the new priority.

This supervisor call is privileged.

Input Parameters

Inline Parameter 1: New priority.

Index Register1: Task control block address.

Example

Program A wants to alter its priority from its current value to 208.

Assembler

```
SVC 36,0
DC XL1'D0'
```

After executing this supervisor call, program A will run at the new priority.

ASYNCHRONOUS TASK READY CHECK

Op Code (hex)	Q-Byte (hex)	R-Byte (hex)
Byte 1	Byte 2	Byte 3
F4	00	25

This instruction accepts a task control block as input and determines whether the task is in an event wait, and, if so, whether the event is completed. Task control block status byte 2 (TCBSTAT2) is checked first to see if the passed task is in an event wait. If the task is not waiting, control is returned to the caller. Index register 1 is used to test for completion of the waited event. If the event is not complete, control is returned to the caller. If the event is complete, a task post (supervisor call 1D) is issued for the posted task control block.

This supervisor call is privileged.

Input Parameters

Q-Byte: Not used; must be zero.

Index Register 1: Task control block address of task to be checked.

Example

Program A wants to have program B's task control block checked to determine if program B may now satisfy a previous wait.

```
Assembler           Object Code

L  @BTCB,XR1
SVC 37,0            F40025
```

PREPARE PRINT BUFFER

Op Code (hex)	Q-Byte (hex)	R-Byte (hex)
Byte 1	Byte 2	Byte 3
F4	00	26

This instruction inserts printer control codes into the print buffer to cause the requested skip and space operations. It also maintains a record of the current logical line in the print input/output block (IOB).

This instruction scans the data to be printed. When it finds more than three contiguous blanks, it compresses those blanks and replaces that data with the relative horizontal print position codes. If it finds a character less than a blank, it replaces that character with a hex FF so that the graphic error action previously set in the printer applies. This scan, compression, and replacement is done within the printer buffer.

If the data is to be routed directly to the printer instead of being spooled, this instruction updates the forms length and current line fields of the associated printer terminal unit block (TUB).

This supervisor call is privileged.

Input Parameters

Q-Byte: Not used; must be zero.

Index Register 1: Points to the address of the input/output block (IOB).

Example

Program A wants to skip to line 1, print a line, and space one line after printing. The IOB address is hex 4544 and the data buffer is at hex 6000.

Program A moves the data to be printed to address hex 6006, sets the skip before printing ($IOBPSKB) and space after printing ($IOBPSPA) fields of the IOB to 1, loads index register 1 with hex 4544, and issues:

Assembler	Object Code
SVC SVCPREP,Q0	F40026

The appropriate control characters are inserted into the data buffer, the data is compressed, and control is returned to program A.

SECTOR ENQUEUE/DEQUEUE

Op Code (hex)	Q-Byte (hex)	R-Byte (hex)
Byte 1	Byte 2	Byte 3
F4	xx	29

The sector enqueue/dequeue supervisor call instruction allows multiple users to share disk sectors while preventing simultaneous access to the same sectors. Index register 1 must contain the address of the disk input/output block (IOB). Sector queue blocks (SQB) are built in the system queue space to protect the sector space.

Upon issuing this supervisor call, any sector queue block which already is associated with the caller's IOB is deleted and all tasks which were waiting on enqueued sectors are posted ready so that their sector enqueue/dequeue supervisor call is reissued.

If the sector queue block count (TCBSQBCT) in the caller's task control block is zero and the caller's task is marked for inquiry pending (bit TCBINQPD in task control block field TCBSTAT3), then the command processor task control block (address hex 0200) is posted ready by posting that task-to-task event control mask (address hex 0250). If the command operation in the IOB (IOB field $IOBCMD) is a read (hex A1) then the caller's task is placed in a general wait in sector enqueue/dequeue failure (bit TCBFWAIT in TCBWMASK field).

If the caller's task has not been waited and the IOB command field is a read operation, then the sector space (sequential sectors starting with the one specified by $IOBDSS extending for as many sectors as are specified by $IOBDN field) is checked for any overlap with any other task's read operation (those tasks which are also issuing this supervisor call) by testing the data saved in the sector queue blocks. If there is any conflict and the wait 0 bit is off, then the program status register false bit is set and control is returned to the caller.

If there is any conflict and the wait 0 bit is on, then the caller's task is placed in a wait on sector enqueue/dequeue failure (bit TCBFWAIT in TCBWMASK field). When the *owning* task of the conflicting sectors issues a sector enqueue/dequeue supervisor call with a nonread operand in the IOB command field ($IOBDCMD) then the caller's task will be posted on TCBFWAIT and will then have a chance to enqueue the required sectors and continue processing.

If there is no sector space conflict with any prior outstanding read operation by any other task, then a sector queue block is built for the caller's task associated with the passed IOB address, and control is returned to the caller.

All existing sector queue blocks are queued to the system queue QHDSQB. Every time a sector queue block is built for a task, the count field in the task control block (TCBSQBCT) is increased by one. Every time a sector queue block is deleted for a task, the count field in the task control block is decreased by one.

This supervisor call is privileged.

Input Parameters

Q-Byte: **Bit**	**Meaning**
0-4,6	Not used; must be zero.
5	(If = 0) IOB address is in translated main storage. (If = 1) IOB address is in nontranslated main storage.
7	(If = 1) Wait for sectors to become available.

Index Register 1: Address of a disk IOB.

Output Values

Program Status Register: False is set if there is a sector conflict and the wait Q-byte bit is not specified.

Example

Program A enqueues four sectors, sequential sectors hex 0077A5 through 0077A8; program B also enqueues these same sectors.

Program A	*Program B*
Index register 1 points to the IOB	Index register 1 points to the IOB (contents same as program A)

IOBDCMD = hex A1 (read op)
IOBDNB = hex 03 (4 sectors)
IOBDSS = hex 0077A5 (starting SS)

Assembler	**Object Code**	**Assembler**	**Object Code**
SVC X’29’,0	F40029	SVC X’29’,0	F40029

Since program A issued the supervisor call first, a sector queue block is built and program A is given back control. Program B is placed in a wait (TCBFWAIT in TCBWMASK field) until program A changes the $IOBDCMD field to a hex A2 and reissues the same supervisor call (SVC X’29’0).

FIXED DISK IOS

Op Code (hex)	Q-Byte (hex)	R-Byte (hex)
Byte 1	Byte 2	Byte 3
F4	xx	40

Index register 1 must point to the input/output block (IOB), which is defined in the disk section of this manual.

This supervisor call is privileged.

Input Parameters

Q-Byte: Bit	Meaning If On
0-2	Not used; must be zero.
3	This event has no translatable main storage requirement.
4	This event is a multiple wait candidate.
5	Input/output block (IOB) is in real storage.
6	Transient area is not refreshable. The current transient is locked into the transient area for the duration of the wait. This takes effect only when issued from instructions executed in the main storage transient area.
7	Wait for requested operation to complete.

Index Register 1: Input/Output block (IOB).

DISKETTE IOS

Op Code (hex)	Q-Byte (hex)	R-Byte (hex)
Byte 1	Byte 2	Byte 3
F4	xx	41

Index register 1 points to the input/output block (IOB) which is defined in the diskette section of this manual.

This supervisor call is privileged.

Input Parameters

Q-Byte: Bit	Meaning If On
0-2	Not used; must be zero.
3	This event has no translatable main storage request.
4	This event is a multiple wait candidate.
5	Input/output block (IOB) is in real storage.
6	Transient area is not refreshable. The current transient is locked into the transient area for the duration of the wait. This takes effect only when issued from instructions executed in the main storage transient area.
7	Wait for requested operation to complete.

Index Register 1: Input/output block (IOB).

WORK STATION PRINTER IOCH

Op Code (hex)	Q-Byte (hex)	R-Byte (hex)
Byte 1	Byte 2	Byte 3
F4	xx	42

Index register 1 points to the input/output block which is defined in the work station section of this manual.

This supervisor call is privileged.

Input Parameters

Q-Byte: **Bit**	**Meaning If On**
0-2	Not used; must be zero.
3	This event has no translatable main storage requirement.
4	This event is a multiple wait candidate.
5	Input/output block (IOB) is in real storage.
6	Transient area is not refreshable. The current transient is locked into the transient area for the duration of the wait. This takes effect only when issued from instructions executed in the main storage transient area.
7	Wait for requested operation to complete.

Index Register 1: Input/output block (IOB).

WORK STATION IOCH

Op Code (hex)	Q-Byte (hex)	R-Byte (hex)
Byte 1	Byte 2	Byte 3
F4	xx	43

Index register 1 must point to the input/output block (IOB) which is defined in the work station section of this manual.

This supervisor call is privileged.

Input Parameters

Q-Byte: **Bit**	**Meaning If On**
0-2	Not used; must be zero.
3	This event has no translatable main storage requirement.
4	This event is a multiple wait candidate.
5	Input/output block (IOB) is in real storage.
6	Transient area is not refreshable. The current transient is locked into the transient area for the duration of the wait. This takes effect only when issued from instructions executed in the main storage transient area.
7	Wait for requested operation to complete.

Index Register 1: Input/output block (IOB).

DATA COMMUNICATIONS IOCH

Op Code (hex)	Q-Byte (hex)	R-Byte (hex)
Byte 1	Byte 2	Byte 3
F4	xx	44

Index register 1 points to the input/output block (IOB), which is defined in the data communications section of this manual.

This supervisor call is privileged.

Input Parameters

Q-Byte: **Bit**	**Meaning If On**
0-1	Not used; must be zero.
2	Hold dispatching until I/O request has been started.
3	This event has no translatable main storage requirement.
4	This event is a multiple wait candidate.
5	Input/output block (IOB) is in real storage.
6	Transient area is not refreshable. The current transient is locked into the transient area for the duration of the wait. This takes effect only when issued from instructions executed in the main storage transient area.
7	Wait for requested operation to complete.

Index Register 1: Input/output block (IOB).

I/O TRANSIENT REQUEST

Op Code (hex)	Q-Byte (hex)	R-Byte (hex)
Byte 1	Byte 2	Byte 3
F4	xx	45

This instruction passes control to the input/output control storage transient area for the current input/output transient. Before this function can be used, the input/output block (IOB) for the specified device must be set up. The input/output block for each device is defined in the section describing the device in this manual.

This supervisor call is privileged.

Input Parameters

Q-Byte: **Bit**	**Meaning If On**
0-2	Not used; must be zero.
3	This event has no translatable main storage requirement.
4	This event is a multiple wait candidate.
5	Input/output block (IOB) is in real storage.
6	Transient area is not refreshable. The current transient is locked into the transient area for the duration of the wait. This takes effect only when issued from instructions executed in the main storage transient area.
7	Wait for requested operation to complete.

Index Register 1: Input/output block.

ACTION CONTROL ELEMENT BUILD AND QUEUE

Op Code (hex)	Q-Byte (hex)	R-Byte (hex)	Inline Parameter List (hex)
Byte 1	Byte 2	Byte 3	Byte 4
F4	xx	4C	xx

The action control element build and queue supervisor call instruction assigns a 16-byte area to be used for the action control element. This action control element is then queued to the passed system queue header (a first-in-first-out queue). The action control element is initialized with the current values of the task as follows:

Bytes	Content
0-1	Address of next action control element (0000 if end of chain)
2-3	Current instruction address register
4-5	Reserved
6-7	Inline parameters 1 and 2
8-9	Inline parameter 3 and type byte
A-B	Current index register 1 value
C-D	Current index register 2 value
E-F	Caller's task control block address

This supervisor call is privileged.

Input Parameters

Q-Byte: **Bit** **Meaning If On**

Bit	Meaning If On
0-2,6,7	Not used; must be zero.
3	This event does not require user main storage. (The task is swappable while the event is active.)
4	This event will satisfy a multiple wait. (This may be used to make translated events satisfy a multiple wait.)
5	The associated event control mask is in real storage.

Inline Parameter 1: Offset of left byte of queue header.

Bit	Meaning
0-6	Displacement of queue header from start of system queue headers.
7	(If = 1) Return address of action control element in index register 2.

Output Values

Index Register 2: Address of the action control element if requested in input.

Example

Program A wants to create an action control element and have it queued to the system queue header at location hex 0148. The address of the action control element is requested.

Assembler	Object Code
SVC X'4C',0 DC XL1'49'	F4004C49

with index register 2 containing hex 0000 and address 0148 containing hex 0000.

After execution of the supervisor call, an action control element is built at location 3478, for example. Location 0148 is set to 3478 and index register 2 is returned, also containing address 3478.

CONTROL STORAGE TRANSIENT SCHEDULER

Op Code (hex)	Q-Byte (hex)	R-Byte (hex)	Inline Parameter List (hex)		
Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
F4	xx	50	xx	xx	xx

This instruction loads control storage transients. The control storage transient to be loaded is identified by a 1-byte transient number given as inline parameter 1 (byte 4). If the transient area is not busy, the requested transient is loaded into the transient area and control is passed to the transient.

Control is returned to the caller after the transient has executed completely.

This supervisor call is nonprivileged.

Input Parameters

Q-Byte: **Bit** **Meaning If On**

0-3,5,7 Not used; must be zero.

4 Parameter list, if needed, is in real storage.

6 Transient area is not refreshable. The current transient is locked into the transient area for the duration of the wait. This takes effect only when issued from instructions executed in the main storage transient area.

Inline Parameter 1: Control storage transient identification.

Inline Parameters 2 and 3: Input to the transient.

Example

Program A wants to call a control storage transient to get the time of day. The transient ID is hex 0A and the input to this transient requires hex 40 in inline parameter 2 and index register 2 must contain the address of a 14-byte timer request block (TRB).

```
Assembler         Object Code

LA  @TRB,XR2
SVC 80,0          F400500A4000
DC  XL'0A'
DC  XL2'4000'
```

When the supervisor call completes, the timer request block contains the current time and date.

TASK WORK AREA ACCESSES

Op Code (hex)	Q-Byte (hex)	R-Byte (hex)	Inline Parameter List (hex)		
Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
F4	xx	51	xx	xx	xx

This instruction is used to access the task work area on disk. Task work area access is a keyed variable-length access method for referencing disk work storage. The key is an offset into the task work area of the sector to be accessed.

Two areas are maintained on disk for each task:
Work station work area–WSWA
Task work area–TWA

Data may be accessed in either of these areas, depending on inline parameter 1. The address of the work station work area or task work area must be given in the task control block field TCBWSWA or TCBTWA.

This supervisor call is privileged.

Input Parameters

Q-Byte: **Bit**	**Meaning If On**
0-5,7	Not used; must be zero.
6	Transient area is not refreshable. The current transient is locked into the transient area for the duration of the wait. This takes effect only when issued from instructions executed in the main storage transient area.

Inline Parameter 1: Type Byte (get/put)

Bit	**Meaning**
0,2-4	Not used; must be zero.
1	Real data addresses.
5	(If = 0) Task work area request.
	(If = 1) Work station work area request.
6	System request.
7	Put request.

Inline Parameter 2: Key (0-59).

Inline Parameter 3: Number of disk sectors (0-255).

Index Register 1: Task control block address (if byte 1 bit 6 = 1).

Index Register 2: Main storage address of data.

Example

Program A wants to read from the task work area. The address in the TCBTWA field is 0F01. Program A wants to read 4 sectors at offset 8 from the task work area. The data address in main storage is hex 2000.

Assembler	**Object Code**
LA X'2000',XR2	C2022000
SVC 81,0	F40051400804
DC XL1'40'	
DC XL1'08'	
DC XL1'04'	

The data located at sequential sector address 000F059 through 000F0C is read into main storage at locations hex 2000 through 23FF.

MAIN STORAGE RELOCATION LOADER

Op Code (hex)	Q-Byte (hex)	R-Byte (hex)	Inline Parameter List (hex)
Byte 1	Byte 2	Byte 3	Byte 4
F4	xx	52	xx

This instruction uses a loader parameter list pointed to by index register 2. It determines the type of request and takes the necessary action as given by inline parameter 1 (byte 4). If relocation is required, the relocation transient is called to perform the relocation after the module has been read into storage.

This supervisor call in nonprivileged.

Input Parameters

Q-Byte: **Bit** **Meaning If On**

Bit	Meaning If On
0-4,7	Not used; must be zero.
5	Parameter list is in real storage.
6	Transient area is not refreshable. The current transient is locked into the transient area for the duration of the wait. This takes effect only when issued from instructions executed in the main storage transient area.

Inline Parameter: Indicates the type of request.

Hex Value	Type of Request
01	Load by relative address: Adds the task loader disk address to the relative address passed in the user's parameter list. The resulting address is the location of the desired module at its link-edit address; control returns to the calling program.
02	Load to address: Reads the module into storage and returns control to the calling program.
04	Fetch: Adds the task relocation factor to the module's link-edit address, and, using the resulting value as the load address, reads the module into storage and passes control to the module's start control address.
06	Fetch to address: Reads the module into storage and passed control to the module's start control address.
0A	System load to address: Updates the task relocation factor and disk address values (in the task's task control block) from the loader's parameter list. Reads the module into storage and returns control to the calling program.
0E	System fetch to address: Updates the task relocation factor and disk address values (in the task's task control block) from the loader's parameter list. Reads the module into storage and passes control to its start control address.

Index Register 2: Loader parameter list. (See *IBM System/34 Data Areas and Diagnostic Aids,* for these parameters.)

Example

Program A wants to load a subroutine from disk into main storage. The sequential sector address of the module is 010342. The parameter list built by program A is:

010342102000200001E102000

The loader supervisor call issued is:

Assembler	Object Code
LA @PL,XR2	
SVC 82,0	F4005202
DC XL1'02'	

Program A will have control returned when the subroutine is loaded at main storage locations 2000 through 2FFF.

Relocation of the subroutine is not necessary since the load address and link address of the subroutine are identical.

Chapter 4. Programming Considerations

INSTRUCTION TIMINGS

Instruction Name	Time in Microseconds
Zero and add zoned	T2 + 1.2L2 + 0.6(L1-L2) + T3
Add zoned decimal	T2 + 1.8L2 + 1.2(L1-L2) + T3
Subtract zoned decimal	T2 + 1.8L2 + 1.2(L1-L2) + T3
Move hex character	T2 + 1.8
Edit (last byte hex 20)	T2 + 1.2L2 + .6L1
Edit (last byte not hex 20)	T2 + 1.2L2 + .6L1 + .8
Insert and test characters	
(last byte significant digit)	T2 + 1.2L1 + .2
(last byte not significant digit)	T2 + 1.2L1 + .6
Move character	T2 + 1.2L
Compare logical characters	T2 + 1.4L
Add logical characters	T2 + 1.8L
Subtract logical character	T2 + 1.8L
Store registers	T1 + 1.4
Load registers	T1 + 1.2
Add to register	T1 + 1.6
Test bits on masked	T1 + 0.8
Test bits off masked	T1 + 0.8
Set bits on masked	T1 + 1.2
Set bits off masked	T1 + 1.2
Move logical immediate	T1 + 0.6
Compare logical immediate	T1 + 0.8
Branch on condition	1.6 (not taken), T1 (taken)
Load address	T1
Jump on condition	1.6 (not taken), 2.0 (taken)
Supervisor call	1.8 + service time
Load program mode register	2.4

Note:

L = length
T1 = 2.4 if direct, 2.0 if indexed.
T2 = 3.6 if direct, 2.8 if indexed, 3.2 if direct/indexed.
T3 = 1.2L1 if recomplemented and result does not equal minus zero, or 1.2 if recomplemented and results equal minus zero, else zero.
L2 = length of operand 2
L1 = length of operand 1

CONDITIONING THE PROGRAM STATUS REGISTER

Machine Instruction	Condition	Binary Overflow	Test False	Decimal Overflow	High	Low	Equal
Zero-add zoned decimal	Set				Operand 2 positive	Operand 2 negative	Operand 2 zero
	Reset				Operand 2 negative	Operand 2 positive	Operand 2 not zero
Add and subtract zoned decimal	Set			Result overflow	Result positive	Result negative	Result zero
	Reset				Result negative or zero	Result positive or zero	Result not zero
Edit	Set				Operand 2 positive	Operand 2 negative	Operand 2 zero
	Reset				Operand 2 not positive	Operand 2 not negative	Operand 2 not zero
Compare logical characters	Set				Operand 1 greater than operand 2	Operand 1 less than operand 2	Operand 1 equal to operand 2
	Reset				Operand 1 not greater than operand 2	Operand 1 not less than operand 2	Operands not equal
Compare logical immediate	Set				Operand 1 greater than immediate data	Operand 1 less than immediate data	Operand 1 equal to immediate data
	Reset				Operand 1 not greater than immediate data	Operand 1 not less than immediate data	Operand 1 not equal to immediate data
Add logical characters	Set	Carry out			Carry out and result not zero	No carry and result not zero	Result zero
	Reset	Reset at start of instruction			No carry or result zero	Carry out or result zero	Result not zero

CONDITIONING THE PROGRAM STATUS REGISTER (continued)

Machine Instruction	Condition	Binary Overflow	Test False	Decimal Overflow	High	Low	Equal
Subtract logical characters	Set				Operand 1 greater than operand 2	Operand 1 less than operand 2	Result zero
	Reset				Operand 1 not greater than operand 2	Operand 1 not less than operand 2	Result not zero
Add to register	Set	Carry out			Carry out and result not zero	No carry and result not zero	Result zero
	Reset	At start of instruction			No carry or result zero	Carry out or result zero	Result not zero
Test bits on			Tested bits not all ones				
Test bits off			Tested bits not all zeros				
Branch or jump on condition	Set						
	Reset		Reset if tested	Reset if tested			
Load register (PSR)	Set	Set if loaded bit 10 on	Set if loaded bit 11 on	Set if loaded bit 12 on	Set if loaded bit 15 off and bit 14 off	Set if loaded bit 15 off and bit 14 on	Set if loaded bit 15 on
	Reset	Reset if loaded bit 10 off	Reset if loaded bit 11 off	Reset if loaded bit 12 off	Reset if bit 15 on, or bit 15 off and bit 14 on	Reset if bit 15 on, or bit 15 off and bit 14 off	Reset if loaded bit 15 off
System reset	Set						Equal set on
	Reset	Binary overflow reset	Test reset	Decimal overflow reset	High reset	Low reset	

Chapter 5. Printer

The IBM System/34 has either an IBM 5211 Printer or an IBM 5256 Printer, or both. Although the 5256 Printer is attached to the system through the work station controller, its functions and programming characteristics are similar to the 5211 printer's. Therefore, both printers are described in this chapter.

PHYSICAL CHARACTERISTICS

The 5211 printer is a line printer with rates of 160 lines per minute (Model 1) or 300 lines per minute (Model 2). These rates were measured using a 48-character set while printing 132 positions to a line with single-line spacing. Other characteristics of this printer are:

- 132 print positions per line
- 6 or 8 lines per inch (25.4 mm)
- 10 characters per inch (25.4 mm)
- Forms width: 3-1/2 inches to 15-1/4 inches
- Forms length: 3 inches to 14 inches
- Character sets of 48, 64, 96, or 128 characters
- Special character-set belts with restricted character sets for better throughput

The 5256 Printer is a serial printer with rates of 40, 80, or 120 characters per second; the rate is determined by the model selected. Other characteristics of this printer are:

- 132 print positions per line
- 6 or 8 lines per inch (25.4 mm)
- 10 characters per inch (25.4 mm)
- Single form/ledger card processing: The maximum size of the forms is 14-1/2 inches wide by 14 inches long. The minimum size is 6 inches wide by 3 inches long.
- Character sets of 96 characters or 128 characters for Katakana

PRINTER KEYS AND LIGHTS

5211 Printer Keys and Lights

The operator's panel on the 5211 printer contains lights and touch keys as shown in Figure 5-1.

Operator Panel Keys

Ready: Turns on the Ready light, sets the printer to a ready state, starts the print belt, and signals the system that the printer is ready to receive commands. This key does not work when either the Interlock or Check light is on, or if the printer controller is not running.

Stop/Reset: Causes the printer to stop and the Ready light to go off. If the printer is printing when the Stop/Reset key is pressed, the current line is completed before the printer stops. This key also resets printer check conditions (Check and Forms lights are reset). In addition, it is used for operator error recovery procedures as described under *Printer Error Recovery,* later in this chapter.

Carriage Restore: Causes (1) the carriage to skip to line 1 of the next form if the Ready, Check, and Interlock lights are off and (2) the horizontal print position to be set to 1. This key operates the same as the Forms Feed command.

Carriage Space: Causes the carriage to advance a single line if the Ready, Check, and Interlock lights are off; this key does not affect the horizontal print position.

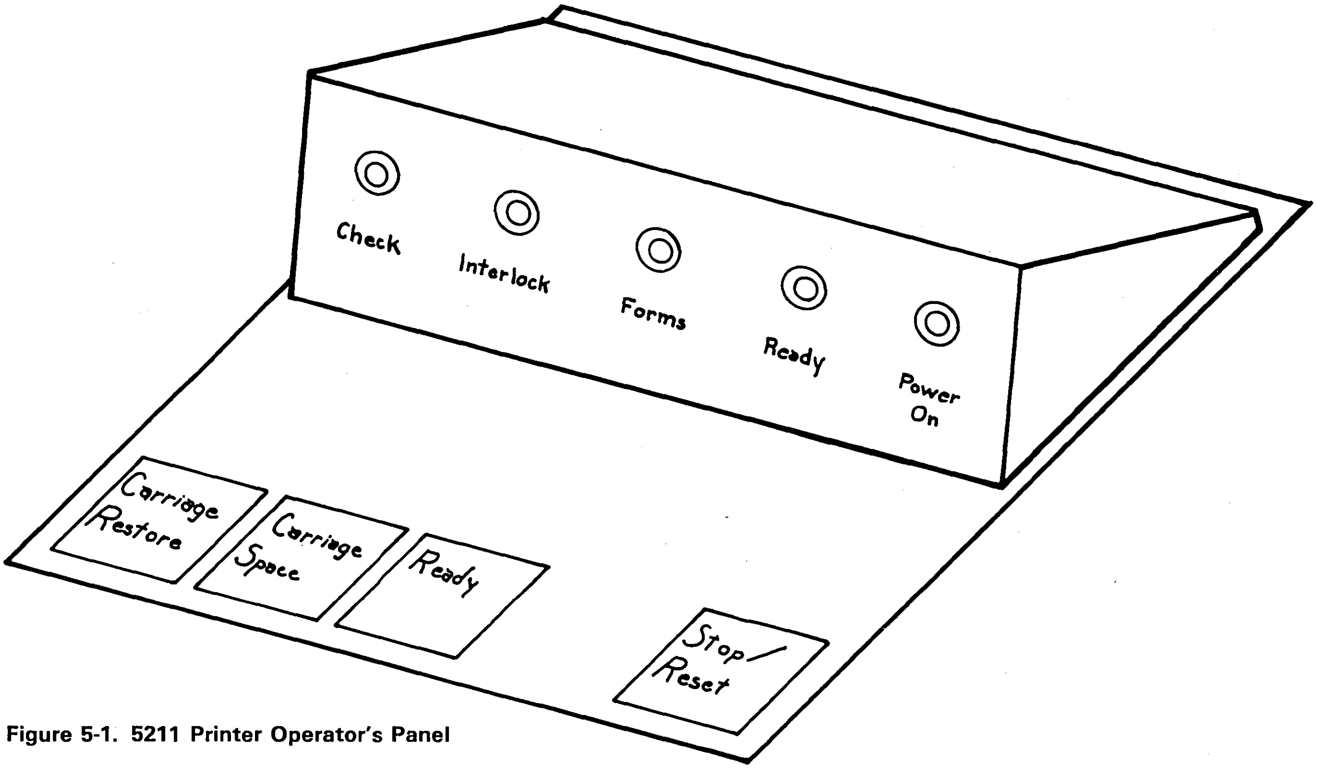


Figure 5-1. 5211 Printer Operator's Panel

Operator Panel Lights

The Check, Forms, and Interlock lights flash on and off when active.

Power On: Indicates that power to the printer is on.

Interlock: Indicates that the printer unit throat is open or the belt cover is not in place.

Forms: Indicates that less than 13 inches (356 mm) of forms remain in the printer below the current print line; more forms must be placed in the printer. If the Forms light is on at the same time as the Check light, a carriage check condition was sensed; this might be a forms jam or loss of vertical position.

Ready: If the Interlock, Forms, and Check lights are off, the Ready light is turned on by the system in response to the Ready key; it indicates that the printer is ready to print.

Check: Indicates that a check condition was sensed by the system and informs the operator that error recovery action is needed. (Also see the description of the Forms light.) If the check condition is corrected, the Check light can be turned off by pressing the Stop/Reset key.

5256 Printer Switches and Lights

The operator's panel on the 5256 printer contains lights and switches as shown in Figure 5-2.

Operator Panel Switches

Power: Turns power on or off to the printer.

Stop: Makes the printer not ready and permits the operator to position the forms. The Stop switch *must* be pressed before positioning the forms by hand, or the system loses control of the vertical positioning. If the Stop switch is pressed while the printer is printing, the current line of printing is completed before the printer stops.

Start: Makes the printer ready to execute commands. The Ready light is on if there is no end-of-forms, not ready, or wire check condition.

Line Feed: Permits a single line feed, but before this switch can be used the printer must be not ready; press the Stop switch to make the printer not ready. This switch operates the same as the line feed command.

The horizontal print position is not affected by the line feed command or the Line Feed switch.

Forms Feed: Permits a series of single line feeds to the first line of the next form. (The horizontal print position is set to 1). The printer must be not ready before the Forms Feed switch will work; press the Stop switch to make the printer not ready.

Line Spacing: Selects the number of lines to be printed per inch; 6 or 8 lines per inch can be selected. If printing at 8 lines per inch, some characters might print over other characters if print wire 8 is used.

Status: Selects online (Normal) or offline (Test) mode. The Cancel position generates a request to cancel the current print operation. The request is reported to the system and a message is displayed on the system console.

When in online mode, the printer can respond to any command from the system if the Ready light is on. When in offline mode, the printer does not respond to any commands from the system. Offline mode is used for problem determination.

Operator Panel Lights

The operator panel has five status lights that show the status of the printer and eight dual-purpose lights that display diagnostic information.

Status Lights

Power On: The power in the printer is on.

Ready: The printer is ready to print data or execute commands from the system.

Attention: Operator action is needed. If the alarm feature is installed, the alarm is also sounded.

Forms: There is some type of forms problem such as forms jammed or end-of-forms.

Unit Check: Indicates a check condition in the printer.

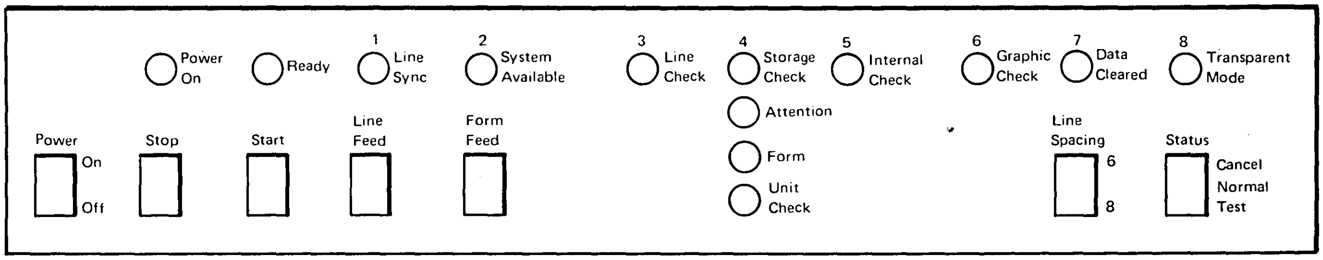


Figure 5-2. 5256 Printer Operator's Panel

Diagnostic Lights

Each of the eight diagnostic lights shows one of two conditions, depending on the position of the Status switch. The following paragraphs describe the meaning of each light when the printer is online (the Status switch is in the Normal position). The device check status conditions are given in parentheses after the light names; these conditions are indicated when the printer is offline (the Status switch is in the Test position).

Line Sync (Wire Check): Signals from the controller caused synchronization with the system. This light is reset by an internally generated signal.

System Available (Slow Speed Check): The printer recognized its own address. This light is reset by an internally generated signal.

Line Check (Fast Speed Check): There was a parity check on data received from the controller. The line parity status is sent to the system and the light is reset by the system.

Storage Check (Left Margin Check): There was a parity check in the printer controller storage. To reset the light, power to the printer must be turned off.

Internal Check (Forms Stopped): There was a parity check in the printer. To reset the light, power to the printer must be turned off.

Graphic Check (Emitter Sequence Check): An unprintable character was sensed. Also, the printer stops printing.

Data Cleared (No Emitters): A clear command has been received from the system. The Data Cleared light will not come on if the printer is ready.

Transparent Mode (Forms Position Lost): Indicates that the printer is in a mode of operation in which the hexadecimal code for each byte of input data is displayed directly above the output of the character.

OPERATIONAL AND PROGRAMMING CHARACTERISTICS

Whether the system has a line printer, a serial printer, or both, the system program handles all printers the same in that each printer attached to the system has an associated terminal unit block (TUB). (The system console and all additional display stations also have an associated TUB.) The TUBs, located in main storage, are chained together on a queue with the TUBs for the printers at the start of the queue. If the line printer is attached to the system, its TUB is the first one on the queue.

The printer queue header byte of the TUB (TUBQHDR) points to a system queue header which in turn points to an action control element and the action control element contains the address of the printers associated input/output block (IOB); this relationship is shown in Figure 5-3. The TUB and IOB are described in the following paragraphs and the action control element is described in the *IBM System/34 Data Areas and Diagnostic Aids*, LY21-0049.

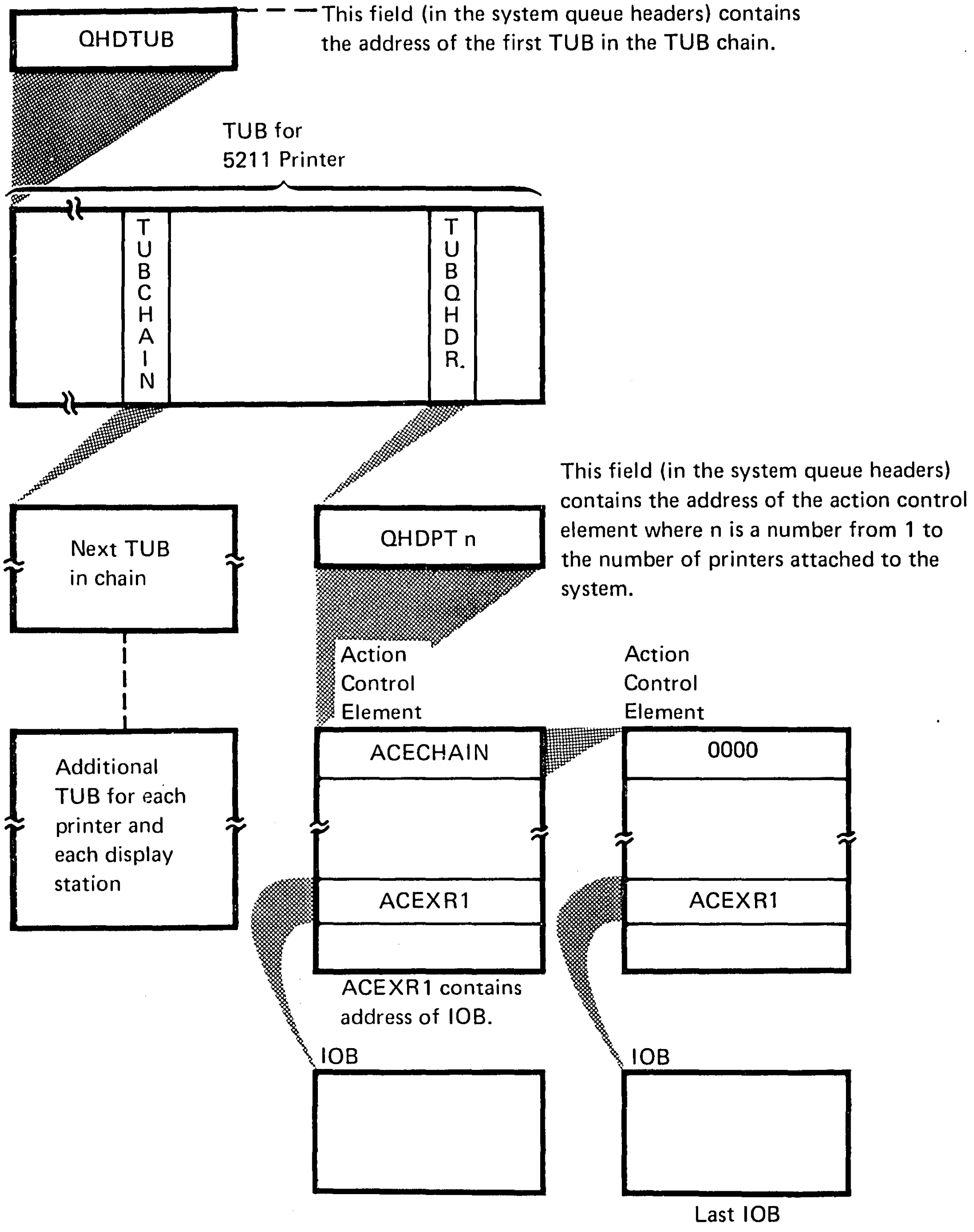

QHDTUB
This field (in the system queue headers) contains the address of the first TUB in the TUB chain.
TUB for 5211 Printer
TUBCHAIN
TUBQHDR.
Next TUB in chain
Additional TUB for each printer and each display station
QHDPT n
This field (in the system queue headers) contains the address of the action control element where n is a number from 1 to the number of printers attached to the system.
Action Control Element
Action Control Element
ACECHAIN
0000
ACEXR1
ACEXR1
ACEXR1 contains address of IOB.
IOB
IOB
Last IOB

Figure 5-3. Relationship of TUB to IOB

Terminal Unit Block

The terminal unit block (TUB), located in main storage, is used by the work station input/output control handler (WSIOCH) routine to identify printers and their associated IOB queue. (WSIOCH is common for display stations and printers). Figure 5-4 shows the format of the TUB as used by the WSIOCH routine.

Displacement of Leftmost Byte in Hex	IBM Program Label	Length in Bytes	Field Description
0	TUBECM	1	Event control mask: This byte is not used by the printers.
1	TUBCOMPL	1	IOB completion code *Hex* *Meaning* 80 Reserved 40 IOB request complete 20 Reserved (must be 0) 10 Reserved 08 Reserved 04 Reserved 02 Reserved 01 Reserved *Note:* Bit 1 is not changed by WSIOCH and it must be set off when opening a printer file and set on when the file is closed to prevent unnecessary control storage activity.
2	TUBFLAG	1	Flag byte *Hex* *Meaning* 80 User defined error recovery procedure 40 Reserved 20 Reserved 10 Reserved 08 Printer is online 04 Reserved 02 Reserved 01 Data is in control storage *Note:* Bit 0 specifies that the system error recovery procedure (ERP) cannot be used. Instead, it allows the reporting of errors to the application program.
3	TUBCMND	1	Device address *Hex* *Meaning* E0 Device address of line printer C0 Device address of work station printer *Note:* The device address (C0 or E0) must be set during initial program load.
4	TUBCMOD	1	Reserved
5	TUBUNIT@	1	Unit address: Address of the serial printer. If this is the TUB for the line printer, the unit address is hex 00.

Figure 5-4 (Part 1 of 4). Printer Terminal Unit Block

Displacement of Leftmost Byte in Hex	IBM Program Label	Length in Bytes	Field Description
6	TUBDATA@	2	Reserved
8	TUBCOUNT	2	Reserved
A	TUBSENS0	1	These are the error status bytes. (See *Printer Status Bytes and Error Recovery Procedures* later in this chapter.)
B	TUBSENS1	1	
C	TUBSENS2	1	
D	TUBSENS3	1	
E	TUBSENS4	1	
F	TUBSENS5	1	
10	TUBTCB@	2	Reserved
The following 10 bytes contain the error recording block; they are used for error recovery procedures (ERP).			
12	TUBCHAIN	2	TUB chain field: Points to the next TUB in the chain; it is set to hex 0000 if this is the end of the chain.
14	TUBDEVID	1	TUB device identification for ERP: Contains a unique code for each printer type on the system; E0 for line printers and E1 for serial printers. *Note:* The device identification identifies the ERP to use. Also, it is set during initial program load.
15	TUBQHDR	1	Queue header pointer for printers: Contains the displacement into the system queue headers for locating the operation queue. *Note:* The queue header pointer is set during initial program load.
16	TUBERPCT	1	ERP control byte *Hex* *Meaning* 80 The ERB is in use waiting for main storage action. 40 Control storage operation complete. 20 Main storage ERB operation complete. 10 Ready response was a second error. 08 Reserved (must be 0).

Figure 5-4 (Part 2 of 4). Printer Terminal Unit Block

Displacement of Leftmost Byte in Hex	IBM Program Label	Length in Bytes	Field Description
16 (continued)			An error is in process if bit 5, 6, or 7 is on and the bits are encoded with the function to be performed. These functions are: *Bits 5 6 7 Function* 1 1 1 Get message identification code (MIC) number from control storage transient area 1 1 0 Perform error recovery 1 0 1 Send message to screen 1 0 0 Waiting for ready condition from printer 0 1 1 Purge message from screen 0 1 0 Not used 0 0 1 Not used
17	TUBERBFG	1	ERB flag byte *Hex Meaning* 80 Put work station identification in the text message. 40 Put 4-byte error code in the last four positions of the text message. 20 No response needed for informational message. 10 Suppress logging. 08 TUB posted complete with error. 04 This ERB is in retry mode. 03 Bit 6 and 7 are encoded as follows: 00 = Not used 01 = Programming error 10 = Hardware error 11 = Resources not available
18	TUBERAID	1	Attention identifier (AID) byte *Hex Meaning* 01 Error pending 00 Attention not requested
19	TUBMIC	2	ERP message identification code
1B	TUBOPTS	1	ERP message options *Hex Meaning* F0 D option was taken to message. 80 Option 0 was selected. 40 Option 1 was selected. 20 Option 2 was selected. 10 Option 3 was selected. 08 Option 0 is allowed. 04 Option 1 is allowed. 02 Option 2 is allowed. 01 Option 3 is allowed.

Figure 5-4 (Part 3 of 4). Printer Terminal Unit Block

Displacement of Leftmost Byte in Hex	IBM Program Label	Length in Bytes	Field Description
The remaining bytes of the TUB contain miscellaneous data.			
1C	TUBSIOCT	2	Start I/O count
1E	TUBERRCT	1	Error retry count
1F	TUBWSID	2	Logical identification of the work station
21	TUBPCFG@	2	SS of printer configuration record
23	TUBTCB	2	TUB owner TCB address and printer queue header
25	TUBAID	1	Reserved
26	TUBCPAID	1	Reserved
27	TUBRESVI	1	Reserved

Figure 5-4 (Part 4 of 4). Printer Terminal Unit Block

Printer Input/Output Block

Printer operations are specified by an input/output block (IOB) located in main storage. Each IOB is queued and sent to the printer by a supervisor call instruction. Multiple operations may be queued by sending multiple supervisor call instructions. See Chapter 3 for a description of the supervisor call instructions.

Each IOB contains an address of a data stream in main storage. This data stream contains the commands to the printer and the data, if any, to be printed. Before using an IOB, the command code (bits 0 through 4), unit address, and queue header displacement bytes must be copied from the correct terminal unit block. Figure 5-5 shows the format of the printer IOB.

Displacement of Leftmost Byte in Hex	IBM Program Label	Length in Bytes	Field Description
0	$IOBPECM	1	Event control mask: This byte is not used directly by the work station input/output control handler but the following bits must be set prior to issuing a supervisor call for a printer. *Hex* *Meaning* 80 No skip bit for general waits. 40 Data adress is real—the supervisor call processor sets up the task control block (TCB) address with this bit. If on = TCB containing real storage. If off = user's TCB. 20 Must be 0.
1	$IOBPSTA	1	Printer completion code *Hex* *Meaning* 80 IOB request active—an operation is in process and is waiting for completion. 40 IOB request complete—the operation is completed. 02 Data transfer complete—data has been transferred to the printer. (This bit is only returned on error conditions.) 01 Error detected—the operation could not be completed because of an error in the printer. *Note:* This bit is turned on if the user's error recovery procedure (ERP) is specified in the terminal unit block (TUB). 00 IOB request inactive.
2	$IOBPFLG	1	IOB flag byte *Hex* *Meaning* 40 Do not allow a 2 option on an error message. 20 Do not issue a message if the message normally needs a 2 or 3 option. 10 Do not log the error. 02 Printer output is to be spooled. 01 Data address is a control storage address.

Figure 5-5 (Part 1 of 3). Printer IOB

Displacement of Leftmost Byte in Hex	IBM Program Label	Length in Bytes	Field Description
3	$IOBPCMD	1	IOB command code *Bits* *0 1 2 3 4 Meaning* 1 1 1 0 0 Line printer is attached. 1 1 0 0 0 Serial printer is attached. *Bits 5 6 7* 0 1 0 Quiesce—requests an interrupt and completion status when the printer operation is complete. 0 0 0 Execute—requests that the data stream associated with this IOB be sent to the printer. *Note:* The IOB is posted complete when the data transfer is complete and a device interrupt indicates another buffer is available.
4	$IOBPMDR	1	IOB command modifier: Command that specifies the operation to be performed. The output and clear commands are supported on all printers. *Hex Meaning* 40 Clear command—required for any permanent printer error resulting in a cancel to clear the printer buffers. *Note:* No data is accepted with the clear command. 27 Output command—causes the information in the data stream to be sent to the printer. The data stream can be from 0 to 256 bytes long.
5	$IOBPUAD	1	Unit address: This is the printer address that is set in the TUB during initial program load (IPL).
6	$IOBPDAT	2	Data address: Provides the logical address of the data to be sent to the printer. (The logical address is the same as the real address if so specified by the event control mask.)
8	$IOBPLNG	2	Length in bytes of the data to be transferred. A length value of more than 256 (maximum length) causes error completion status.

Figure 5-5 (Part 2 of 3). Printer IOB

<table>
<tr><th>Displacement of Leftmost Byte in Hex</th><th>IBM Program Label</th><th>Length in Bytes</th><th>Field Description</th></tr>
<tr><td colspan="4">The following bytes are used by the prepare print buffer supervisor call to insert control characters into the data buffer and to maintain a record of the current print line.</td></tr>
<tr><td>A</td><td>$IOBPCTL</td><td>1</td><td>Control byte
<i>Hex Meaning</i>
80 Alignment requested
40 Print operation</td></tr>
<tr><td>B</td><td>$IOBPSKB</td><td>1</td><td>The line number to skip to before printing.</td></tr>
<tr><td>C</td><td>$IOBPSPB</td><td>1</td><td>The number of lines to space before printing.</td></tr>
<tr><td>D</td><td>$IOBPSKA</td><td>1</td><td>The line number to skip to after printing.</td></tr>
<tr><td>E</td><td>$IOBPSPA</td><td>1</td><td>The number of lines to space after printing.</td></tr>
<tr><td>F</td><td>$IOBPWKA</td><td>1</td><td>Work area.</td></tr>
<tr><td>10</td><td>$IOBPTCB</td><td>2</td><td>Task control block (TCB) address: Address locates the address translation register (ATR) for the requesting task.
<i>Note:</i> Bit 1 of the event control mask determines how the supervisor call processor sets this address.</td></tr>
<tr><td>12</td><td>$IOBPSQD</td><td>1</td><td>Printer queue header displacement: Contains the displacement into the system queue headers of the printer queue for this printer.</td></tr>
<tr><td>13</td><td>$IOBPTUB</td><td>2</td><td>Address of the terminal unit block.</td></tr>
<tr><td colspan="4">The following bytes are used by the prepare print buffer supervisor call to validate the skip and space values and to maintain a record of the current print line.</td></tr>
<tr><td>15</td><td>$IOBPFML</td><td>1</td><td>Forms length (lines per page).</td></tr>
<tr><td>16</td><td>$IOBPCLN</td><td>1</td><td>Current line number.</td></tr>
<tr><td>17</td><td>$IOBPLEN</td><td>1</td><td>Length of the printer IOB.</td></tr>
</table>

Figure 5-5 (Part 3 of 3). Printer IOB

Printer Output Data Stream

The output data stream contains all print data and commands to the printer. The data stream (1) is limited to 256 bytes since this is the size of the printer receive buffers, (2) is free-form; that is, commands can appear anywhere in it, and (3) must be prepared by the program before the IOBs are put on the queue. The commands, described in the following paragraphs and in Figure 5-7, have hexadecimal values from 00 to 3F, and the print data characters have hexadecimal values from 40 to FF.

Note: Program transparency to a printer type can be guaranteed only when the commands used are common to both the serial and line printers.

Printer Commands

The printer commands control carriage operations and supply formatting information for forms length and chain image. For a better understanding of the following command descriptions, see Figure 5-6, which shows the format of a printer form. View the form as a *presentation surface* on which a *presentation position* (the print position after an executed command) can be moved.

A summary of all the commands is given in Figure 5-7. This figure gives the hexadecimal code for each command, the command description, and the availability of the commands by printer type.

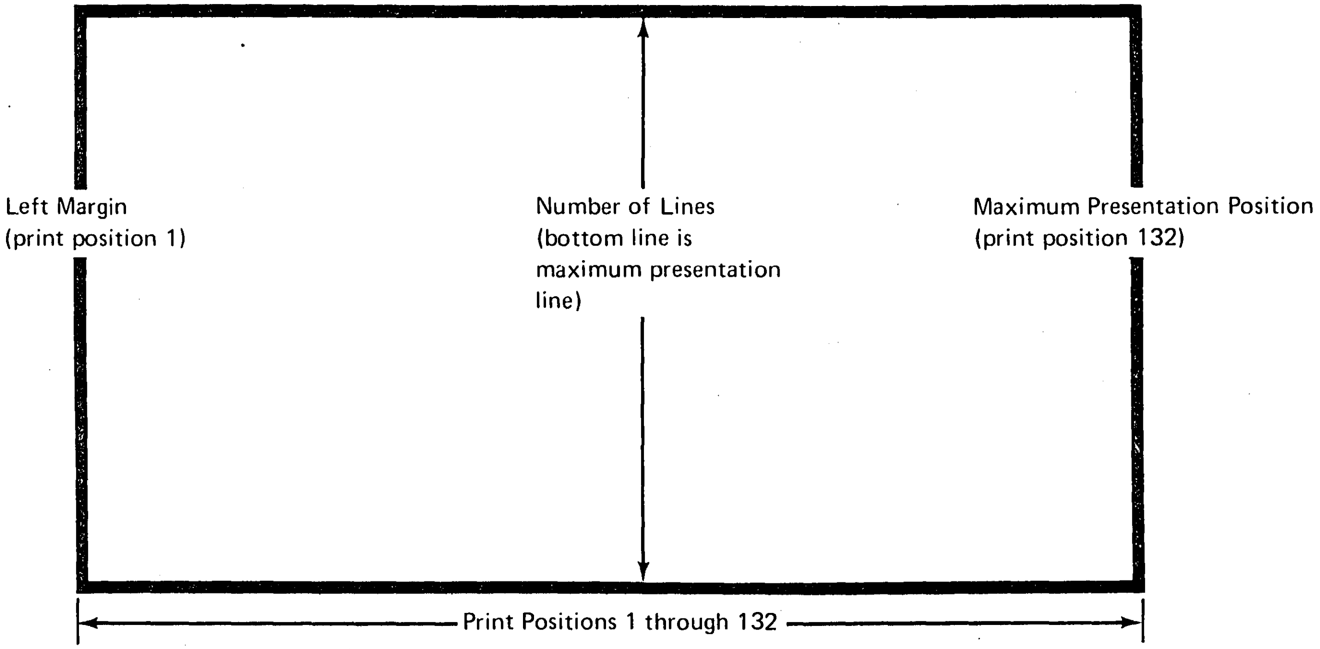


Figure 5-6. Format of Printer Form

Hexadecimal Code	Command Description	Available on Printer Type
00	Null: No character is printed and no function is performed.	Serial
0C	Forms feed: Moves the print position to the first position of the next logical page. *Note:* Although it is not a restriction, it is suggested that all new pages be entered with the forms feed command to assist spool intercept in its numbering of pages.	Serial and line
0D	Carriage return: Moves the print position to the first position of the same line. If the current print position is the first position of a line, a no-op occurs.	Serial and line
15	New line: Moves the print position to the first position of the next line. If the current print position is on the last line of a page, the print position is moved to the first position of the first line on the next page.	Serial and line
1E	Interchange record separator: Same as New line.	Serial
25	Line feed: Moves the print position to the same print position of the next line. If the print position is on the last line of a page, it is moved to the same print position of the first line on the next page.	Serial
2B	Format: Hex 2B is the format character that defines the start of a formatting data stream. The format character is always followed by a function byte (C1, C2, C7, or C8) that identifies the format type. The function byte is followed by a count byte which specifies the length of the remaining data stream (including the count byte). *Note:* Formats must be established before any print or carriage operation. Formats not used by a printer will cause an invalid SCS (standard character string) parameter check.	See the following four format descriptions.
2BC1nnhh	Set horizontal format: Defines the maximum horizontal print position, where: nn = number of bytes remaining in the data stream plus 1. This value is 02 when hh is specified but valid values are 01 and 02. If nn is 01, hh defaults to 132. An invalid value causes a default of 132 for hh and an SCS parameter check. hh = maximum horizontal print position up to, and including, 132. An invalid value causes a default of 132 and an SCS parameter check.	Serial

Figure 5-7 (Part 1 of 3). Printer Commands Description

Hexadecimal Code	Command Description	Available on Printer Type
2BC2nnvv	Set vertical format: Defines the forms length, where: nn = number of bytes remaining in the data stream plus 1. This value is 02 when vv is specified but valid values are 01 and 02. If nn is 01, vv defaults to 01. An invalid value causes a default of 01 for vv and an SCS parameter check. vv = forms length up to and including 255. An invalid value causes a default of 01 and an SCS parameter check. *Note:* When the forms length is set, the current line is set to 1.	Serial and line
2BC7nncsi	Set chain image: Loads the character set image of the line printer, where: nn = number of bytes remaining in the data stream plus 1. csi = character set image representing the sequence of print characters as they appear on the print belt.	Line
2BC8nngguu	Set graphic error action: Defines the action to be taken when a character that is not defined in the character set image is sensed, where: nn = number of bytes remaining in the data stream plus 1. This value is 03 when gg and uu are specified but valid values are 01 through 03. The defaults apply when a value of 01 is specified. gg = default graphic. If gg is not specified, it defaults to a blank (hex 40). *Note:* For the line printer, all defaults are a blank character. For the serial printer, the default graphic is the character specified. However, for any specified value below hex 40, the default is a blank. uu = unprintable character option of: 00—the default of 01 is set 01—neither halt nor return status 02—neither halt nor return status for expanded character set 03—halt and return status 04—halt and return status for expanded character set *Note:* Invalid values for nn and/or uu cause the defaults to be set, and cause an invalid SCS parameter check.	Serial and line
2F	Bell: Turns on the Attention indicator to indicate that operator action is needed and, if the audible alarm feature is installed, it sounds the alarm.	Serial

Figure 5-7 (Part 2 of 3). Printer Commands Description

Hexadecimal Code	Command Description	Available on Printer Type
34	Presentation position: Hex 34 is the character that defines the start of a presentation (print) position data stream. This character is always followed by a function byte (C0, C4, C8, or 4C) that identifies the type of move. The function byte is followed by a value byte that specifies a column or line number (in the absolute sense) or an incremental change from the current print position (in the relative sense). *Note:* Invalid function and/or value bytes cause an invalid SCS (standard character string) parameter check. Also, no carriage or forms movement operations are performed.	See the following four presentation position descriptions.
34C0ah	Absolute horizontal position: Moves the print position to the horizontal position specified by the ah parameter. Valid values for ah include any positive number up to, and including, 132. *Note:* If the value specified by ah is less than the current horizontal position, a carriage return is performed and the line is printed again to the specified position. An ah value of zero causes a no-op; a value greater than the end of the line causes an invalid SCS parameter check.	Serial
34C4av	Absolute vertical position: Moves the print position to the line specified by the av parameter. Valid values for av include any positive number up to, and including, the forms length. The horizontal position does not change. *Note:* If the value specified by av is less than the current line, the forms are moved to the specified line of the next page. An av value of zero causes a no-op; a value greater than the forms length causes an invalid SCS parameter check.	Serial and line
34C8rh	Relative hoizontal position: Moves the print position horizontally from its present position the number of positions specified by the rh parameter. Valid values for rh include any positive number up to, and including, the maximum print position. *Note:* A rh value of zero causes a no-op; a value greater than the end of the line causes an invalid SCS parameter check.	Serial and line
344Crv	Relative vertical position: Moves the print position vertically from its present position the number of lines specified by the rv parameter. Valid values for rv include any positive number up to, and including, the number for the last line on the form.	Serial

Figure 5-7 (Part 3 of 3). Printer Commands Description

Form Feed (OC)

This command moves the presentation position to the top line and left margin of the next page as specified by the maximum print line parameter which is set by the set vertical format operation as described under *Format (2B)*. If it is not set, the maximum print line is assumed to be one, and the presentation position moves to the left margin of the next line.

Programming Note: Use the form feed command to move the presentation position to a new form because this command is used for page numbering by spool intercept. Do not use the new line command or the absolute vertical parameter when moving the presentation position from one form to the next.

Carriage Return (OD)

This command moves the presentation position to the left margin of the same line. If the current presentation position equals the left margin, the carriage is not moved.

New Line (15)

This command moves the presentation position to the left margin of the next line. If a sequence of print characters attempts to cause the presentation position to go beyond 132 (the maximum presentation position), an automatic new line is generated.

Format (2B)

The format character (hex 2B) specifies the start of a formatting data stream. It is used with one of the function bytes described below (hex C1, C2, C7, and C8). The function byte is followed by a count byte that specifies the number of bytes remaining in the formatting data stream (including the count byte). All of the printer formats must be assembled before any print or carriage operation.

Set horizontal format (2BC1): Specifies the horizontal format for the forms width. The format code of 2BC1 is followed by a count byte of either hex 01 or hex 02. If the count byte is hex 02, a forms width byte follows the count byte. If the count byte is hex 01, the forms width byte has a default value of 132.

Set vertical format (2BC2): Specifies the vertical format for the forms length. The format code of hex 2BC2 is followed by a count byte of either hex 01 or hex 02. If the count byte is hex 02, a forms length byte follows the count byte. If the count byte is hex 01, the forms length byte has a default value of 1.

Set chain image (2BC7): Loads the character set image of the line printer. The format code of hex 2BC7 is followed, in order, by:

1. A count byte of L+1, where L is the length of the chain image.
2. The character set image in the sequence that the characters appear on the print belt.

Set graphic error action (2BC8): Causes a substitute character to be printed when an unprintable character is sensed in the data stream. The format code of hex 2BC8 is followed, in order, by:

1. A count byte of hex 03. (See note below.)
2. The character (blank) substituted for the unprintable character.
3. An error stop control byte of either hex 01 (no stop on unprintable characters and no status returned) or hex 03 (unit check set and status returned). A value of hex 00 causes a default of no stop on unprintable characters.

Note: If the count byte is hex 01, the defaults are set with blank character substitution and no stop on unprintable characters. A count byte of hex 02 permits character substitution but the error stop control byte defaults to no stop on unprintable characters.

Presentation Position (34)

This command, used with four different function parameters, moves the presentation position as specified by the parameters. Each function parameter follows the command in the data stream; they are described below. A value parameter follows the function parameter in the data stream; it is a 1-byte number that describes either a position or line number.

The four valid function parameters are the absolute horizontal position (hex C0), the absolute vertical position (hex C4), the relative horizontal position (hex C8), and the relative vertical position (hex 4C). If any value other than hex C0, C4, C8, or 4C is sensed, an invalid SCS (standard character string) parameter is indicated.

When an absolute horizontal position is specified, the presentation position is moved to the print position specified by the value parameter. This parameter is valid for any move to (but not past) the end of the line.

When an absolute vertical position is specified, the presentation position is moved to the line specified by the value parameter. This parameter is valid for any value that does not exceed the maximum presentation line. A value that is less than the current line will cause the presentation position to be moved to the specified line on the next page. The horizontal position is not affected.

When a relative horizontal position is specified, the presentation position is moved relative to the current position, by the number of positions specified in the value parameter. This parameter is valid for any move to (but not past) the end of a line.

When a relative vertical position is specified, the presentation position is moved relative to the current position, by the number of lines specified in the value parameter. This parameter is valid for any move down to (but not past) the last line on the form.

Printer Status Bytes and Error Recovery Procedures

When an error is sensed at the end of an operation, the terminal unit block is posted with status and error bytes indicating printer conditions, and processing of the IOB queue stops. If the user's error recovery procedure is used, the first IOB on the queue is posted complete with error. However, if the system's error recovery procedure is used, the control processor is called to process the error. In order to determine the cause of the error, the control processor checks the status bytes in a priority sequence (see Figures 5-8 and 5-10 for the priorities).

Because the 5256 printer is attached to System/34 as a work station printer, it needs an additional level of recovery not needed by the 5211 printer. All possible error conditions associated with the twinaxial interface (the 5256 printer is connected to System/34 through a twinaxial cable), must be corrected before the printer errors are checked. These possible error conditions are common for the 5256 printer and all display stations and are described in Chapter 7.

The status bytes and error recovery procedures for the 5211 printer are given in Figures 5-8 and 5-9 respectively; for the 5256 printer, they are given in Figures 5-10 and 5-11, respectively.

Status Byte	Bit	Test Priority	Bit Name	Bit Description	Suggested Recovery Action (See Note 1.)
0	0	2	Controller unit check	Hardware parity check or controller time-out. *Note:* If this bit is on, status bytes 1 through 5 contain all zeros. In addition, the status of this bit determines the contents of bits 1 through 7 of status byte 0.	5
0 (with bit 0 off)				When bit 0 is off, bits 1 through 7 mean the following:	
	1	16	Unprintable character	A print character not defined in the belt image was sensed.	1
	2	9	Hammer echo check	Incorrect printing, or no printing when printing should have occurred. (Also see description of status bytes 4 and 5.)	1
	3	28	Not ready	Printer is not ready to print.	4
	4	13	Belt sync check	Printer is out of sync; printing might be in error.	1
	4	15	Belt speed check	Belt either failed to start or it stops.	1
	6	14	Belt up to speed check	Belt failed to get up to speed in required time.	1 or 6 (See Note 2.)
	7	5	Any hammer on check	A print hammer is on when it should not be. Power to the printer is de-activated.	1
0 (with bit 0 on)				When bit 0 is on, bits 1 through 7 mean the following:	
	1	N/A	Not used		N/A
	2-3	N/A	Controller unit check decode	00 = Controller time-out 01, 10, or 11 = Hardware parity check	N/A
	4	N/A	Not used		N/A
	5-6	N/A	Speed select jumpers	00 = Jumpers not correctly placed on adapter card 01 = 300 lines-per-minute printer 10 = Jumpers not correctly placed on adapter card 11 = 160 lines-per-minute printer	N/A
	7	N/A	CE sense bit		N/A

Figure 5-8 (Part 1 of 3). 5211 Printer Status Bytes

Status Byte	Bit	Test Priority	Bit Name	Bit Description	Suggested Recovery Action (See Note 1.)
1	0	22	End of forms	Less than 13 inches of forms remain in printer.	3
	1	11	Forms jam	Forms fail to move in last 10 to 22 lines; overprinting probably occurred.	2
	2	23	Throat open	The paper path throat is open or the belt cover is not in place . (Interlock light is on.)	1 or 4 (See Note 3.)
	3	24	Printer busy too often	Printer was busy too often in a single print operation; that is, the 'printer busy' line was active more than 3 times.	1
	4	25	Printer busy too long	The 'printer busy' line was on more than 3 seconds during a single print operation.	1
	5	26	Ribbon check	The ribbon is moving too slow.	1
	6	3	Cable interlock	A cable is unplugged in either the attachment board, cable tower, printer unit, or printer console.	1
	7	27	Hammer bus out parity check	Printer sensed even parity on the bus out lines during print time.	1
2	0	7	Not powered on	No power to the printer. (Power light is off.)	4
	1	17	Data transfer check	Data byte from the system was lost or an extra data byte was sensed.	5
	2	19	Data stream reject	An invalid data stream was sent to the printer.	5
	3	N/A	Not used		N/A
	4	21	Invalid SCS parameter	The parameter byte that follows an SCS control character was not recognized.	5
	5	20	Invalid SCS command	An undefined control character was sensed in the data stream.	5
	6	18	Invalid IOB	The data stream length in the IOB exceeded 256 bytes, or an invalid command code or command modifier was sensed.	5
	7	4	Printer power check	The printer lost power unexpectedly. (Power light is off.)	1 (See Note 4.)

Figure 5-8 (Part 2 of 3). 5211 Printer Status Bytes

Status Byte	Bit	Test Priority	Bit Name	Bit Description	Suggested Recovery Action (See Note 1.)
3	0	6	CE switch on	A CE switch on the printer is on.	4
	1	N/A	Lines per inch	If on, 8 lines per inch; if off, 6 lines per inch.	N/A
	2-3	N/A	Model attached	*Bit 2* *Bit 3* 0 0 Model 1 (160 lpm) 0 1 Model 2 (300 lpm)	N/A
	4	8	Fire tier check	Fire tier lines are out of sequence.	1
	5	12	PSS emitter check	Print subscan emitters failed.	1
	6	10	Carriage check 2 (carriage speed check)	A single-space operation did not complete in 34 milliseconds, the time needed to keep printing at the rated speed. This is not an error unless it occurs 3 times on a single page.	2
	7	10	Carriage check 1 (carriage sync check)	A carriage emitter pulse failed to occur when expected, or occurred when not expected.	2
4	0-7	1	Hex FF	Status bytes 0 through 3 contain the error status.	N/A
	0-7	1	Hex 04	Invalid storage page or I/O buffer boundary was used for the print buffer.	6
	0-7	1	First failing hammer	If a hammer echo check occurs (bit 2 of status byte 0 is on), this byte contains, in hex, the number of the first failing hammer.	N/A
5	0-7	1	Number of failing hammers	If a hammer echo check occurs (bit 2 of status byte 0 is on), this byte contains, in hex, the number of failing hammers.	N/A

Notes:

1. The suggested recovery actions are described in Figure 5-9.
2. Recovery action 6 requires no action from the operator; the printer recovers itself.
3. Perform recovery action 1 if the throat was opened while printing; otherwise perform recovery action 4.
4. Turn the printer off, and then go on. Then perform recovery action 1.

Figure 5-8 (Part 3 of 3). 5211 Printer Status Bytes

Suggested Recovery Action	Lights (see note)	Type of Error	Error Description and Recovery Action
1	Check	Print check	An error occurred while printing. Operator action: 1. Correct error condition. 2. Press Stop/Reset to clear the check condition (Check light goes off). 3. Press Ready (Ready light comes on) Printer recovery: After Ready is pressed, the printer tries the operation again; if not successful, more operator action is needed.
2	Check and Forms	Carriage check	Either the carriage moved too slowly, or it moved when not expected, or it did not move when expected. Operator action: 1. Press Stop/Reset to clear the check condition (Check and Forms lights go off). 2. Press Carriage Restore to set the current line counter to 1. 3. Manually align the forms to line 01 of the next form to be printed. 4. Press Ready. Print recovery: After Ready is pressed, the printer skips to the line it was headed for when the error occurred unless the carriage moved when not expected.
3	Forms	End of forms	The printer is out of forms. Operator action: 1. Press Stop/Reset to clear the check condition (Forms light goes off). 2. Put more forms in the printer. 3. Press Ready. *Note:* One more form can be printed before putting new forms in the printer; press Stop/Reset to clear the check condition and then press Ready. Printer recovery: Normal operation continues after pressing Ready.

Figure 5-9 (Part 1 of 2). 5211 Printer Error Recovery Procedures

Suggested Recovery Action	Lights (see note)	Type of Error	Error Description and Recovery Action
4	None or Interlock	Not ready	The printer is not ready. Operator action: 1. Press Ready, or 2. If the Interlock indicator is on, correct error condition and then press Ready. Printer recovery: Any operations loaded into the print buffers begin executing. *Note:* If the printer does not respond after pressing the Ready key, the printer controller is not operational. System IPL is needed.
5	None or Ready	Program check	System or programming error occurred. Operator action: Cancel the job from the system console. Printer recovery: None.
6	Ready	Soft error	An error was sensed but recovery was automatic.
Note: The *Lights* column identifies the lights that come on for each error condition.			

Figure 5-9 (Part 2 of 2). 5211 Printer Error Recovery Procedures

Status Byte (See Note 1.)	Bit	Test Priority	Bit Name	Bit Description	Suggested Recovery Action (See Note 2.)
0	2	(See Note 3.)	Work station controller command reject	When this bit is on, status byte 4 (IOB 14) contains a code that gives the reason for the error. If status byte 4 contains: —Hex 01, the data stream was directed to a printer when the print buffer in the work station controller was full. —Hex 04, the data stream was directed to a printer when the printer needed initialization.	7
2	2	13 (See Note 4.)	Device not available	The printer is not ready. This bit may be on by itself if the printer has not been made ready, or it may be on with another bit to indicate a hard error. The other bits that may be on at the same time that this bit is on are bits 0 through 7 of status byte 5 and bit 5 of status byte 4.	4
2	3	None	Outstanding status	Indicates a printer error condition with status posted in status bytes 4 and 5.	None
3	0-7	(See Note 3.)	Hex FF	The work station controller sensed an error while processing. The status is contained in status bytes 0, 1, 2, 4, and 5.	None
3	0-7	(See Note 3.)	Hex D1	The Cancel key was pressed. A message is displayed on the screen offering cancel options. Select option 0 if you want to ignore the cancel.	None
3	0-7	(See Note 3.)	Hex 04	An invalid storage page or boundary was sensed by the work station input/output control handler.	5
4	0	10	Invalid SCS command	A character between hex 00 and hex 3F in the data stream was not recognized.	5
4	1	11	Invalid SCS parameter	One of the parameter bytes following the SCS command byte in the data stream was not recognized.	5
4	5	3 (15) (See Note 4.)	Printer mechanism not ready	The printer power is off or an undervoltage condition exists. *Note:* If the printer automatically recovers from this error, the print head moves to the left margin and prints the character (and all remaining characters on the print line) that was being printed when the error occurred.	1 or 6 (See note 5.)

Figure 5-10 (Part 1 of 3). 5256 Printer Status Bytes

Status Byte (See Note 1.)	Bit	Test Priority	Bit Name	Bit Description	Suggested Recovery Action (See Note 2.)
4	6	1	End of forms	The end of forms was sensed. More forms must be loaded in the printer.	3
4	7	12	Graphic error (unprintable character)	A hexadecimal value between hex 40 and hex FF not defined in the printer character set was sensed.	1
5	0	2 (15) (See Note 4.)	Wire check	Print wires were on too long while printing. The 'printer mechanism not ready' bit also comes on.	1 or 6 (See Note 5.)
5	1	20	Slow speed check	The print emitters are occurring too slow. The printer continues to print but at a slower speed.	6
5	2	7 (17) (See Note 4.)	Fast speed check	The print emitters are occurring too fast. The print coils may overheat. *Note:* If the printer automatically recovers from this error, the print head moves to the left margin and prints the character (and all remaining characters on the print line) that was being printed when the error occurred.	1 or 6 (See Note 5.)
5	3	6 (16) (See Note 4.)	Emitter sequence check	The print emitters did not occur in the proper sequence. This usually indicates that the head moved in the wrong direction.	1 or 6 (See Note 5.)
5	4	8 (18) (See Note 4.)	No emitters	No emitters were sensed while the head was moving. *Note:* If the printer automatically recovers from this error, the print head moves to the left margin and prints the character (and all remaining characters on the print line) that was being printed when the error occurred.	1 or 6 (See Note 5.)
5	5	9 (19) (See Note 4.)	Overrun check	The emitter pulses occurred faster than the work station controller could handle them.	1 or 6 (See Note 5.)
5	6	5	Forms stopped	The carriage feedback to the system failed when the forms were advanced. The forms are probably jammed.	2
5	7	4	Forms position check	The carriage feedback to the system indicates that the forms moved more than expected since the last carriage operation. The work station controller has lost the position of the current print line.	2

Figure 5-10 (Part 2 of 3). 5256 Printer Status Bytes

Notes:

1. Most of the bits in status bytes 0 through 2 for the printer are common with the same bits for the display station. The common bits are:
   - Bits 0, 1, 3, 4, 5, and 7 of status byte 0
   - Bits 1 through 5 and bit 7 of status byte 1
   - Bits 0, 1, and 4 through 6 of status byte 2

   See Figure 7-13 for a description of these bits.
2. The suggested recovery actions are described in Figure 5-11.
3. Status byte 3 has the highest test priority followed by the bits in status bytes 0 through 2. The priority of the bits in status bytes 0 through 2 are given in Figure 7-13. After all the bits in status bytes 0 through 2 have been tested, the bits in status bytes 4 and 5 are tested in the priority that is given in the *Test Priority* column.
4. If bit 2 of status byte 2 is off, bit 5 of status byte 4, and bits 0 and 2 through 5 of status byte 5 are tested for a soft error. The test priority for the soft errors is given in parentheses.
5. No operator action is needed for recovery action 6; the printer recovers automatically.

Figure 5-10 (Part 3 of 3). 5256 Printer Status Bytes

Suggested Recovery Action	Lights (see note)	Type of Error	Error Description and Recovery Action
1	Unit Check	Print check	An error occurred while printing. Operator action: 1. Correct the error condition. 2. Press Stop to clear the check condition (Unit Check light goes off). 3. Press Start (Ready light comes on). Printer recovery: Automatically tries operation again; if not successful operator action is needed.
2	Unit Check and Forms	Carriage check	Either the carriage moved when not expected, or it did not move when expected. Operator action: 1. Press Stop to clear the check condition (Unit Check and Forms lights go off and the print head moves to the left margin). 2. Press Form Feed to set the current line counter to 1. 3. Manually align the forms to line 1 of the next form to be printed. 4. Press Start. Printer recovery: After Start is pressed, the printer skips to the line it was headed for when the error occurred.
3	Forms	End of forms	The printer is out of forms. Operator action: 1. Press Stop to clear the check condition (Forms light goes off). 2. Put more forms in the printer. 3. Press Start. Printer recovery: Normal operation continues after pressing Start.
4	None	Not ready	The printer is not ready. Operator action: Press Start. Printer recovery: Any operations loaded into the print buffer begin executing.

Figure 5-11 (Part 1 of 2). 5256 Printer Error Recovery Procedures

Suggested Recovery Action	Lights (see note)	Type of Error	Error Description and Recovery Action
5	Ready	Program check	System or programming error occurred. Operator action: Cancel the job from the system console; the printer remains ready. Printer recovery: None.
6	Ready	Soft error	An error was sensed but recovery was automatic.
7	None	Twinaxial interface check	An error occurred on the twinaxial interface to the work station. The printer is reset and data has been lost. Operator action: Cancel the job from the system console. Printer recovery: None.
Note: The Lights column identifies the lights that come on for each error condition.			

Figure 5-11 (Part 2 of 2). 5256 Printer Error Recovery Procedures

Chapter 6. Disk

The disk which is inside of the 5340 System Unit, has either 8.6 or 13.2 megabytes of storage. A second disk can be installed to give a total of 27.1 megabytes of storage. The specifications for the disk are shown in Figure 6-1.

The disk has two areas of data tracks on the front side and one more area on the back side. There are three data heads installed on a moving actuator, one data head for each data area. There is one servo head installed on the actuator, the servo head uses a separate area of servo tracks on the back side of the disk. The servo head guides the actuator during a seek and during track following.

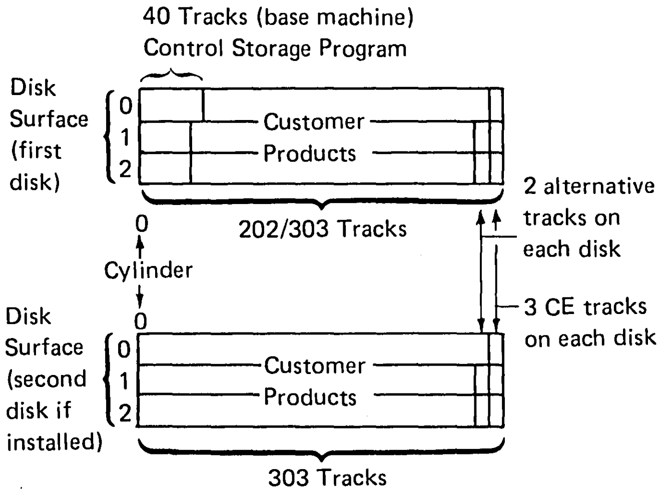
Item	**Speed or Size**
Rotational speed	2964 (±3.0%) RPM
Average rotational delay or latency	10.1 milliseconds
Average seek time (excluding latency) over one-third of the disk:	
9.1 megabytes	35 milliseconds
13 megabytes	40 milliseconds
Capacity	
Sectors per track	60
Bytes per sector	256
Bytes per track	15,360
Tracks per cylinder	3
Bytes per cylinder	46,080
Cylinders (9.1)	202
Capacity (9.1)	9,169,920 bytes
Cylinders (13.7)	303
Capacity (13.7)	13,777,920 bytes
Data rate	889,000 bytes/sec or 1.13 μs/byte or 141 ns/bit

ONU089

Figure 6-1. Disk Speed and Size

DISK SURFACE

Cylinder 0 is the inside edge of the data area and the CE cylinder is on the outer edge of the data area for each head. The tracks are shown in Figure 6-2 and Figure 6-3.



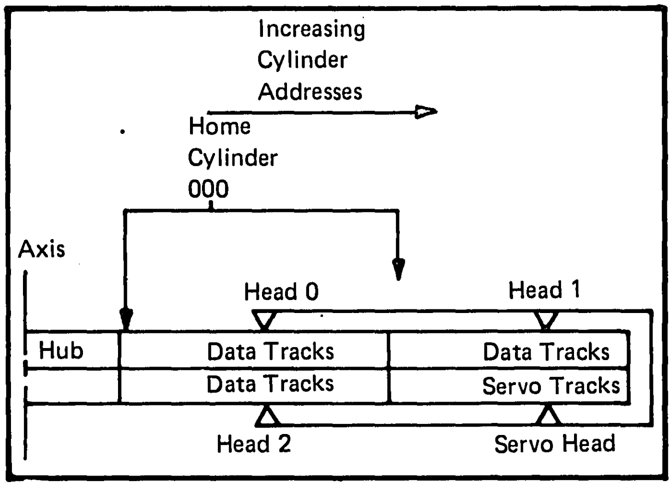
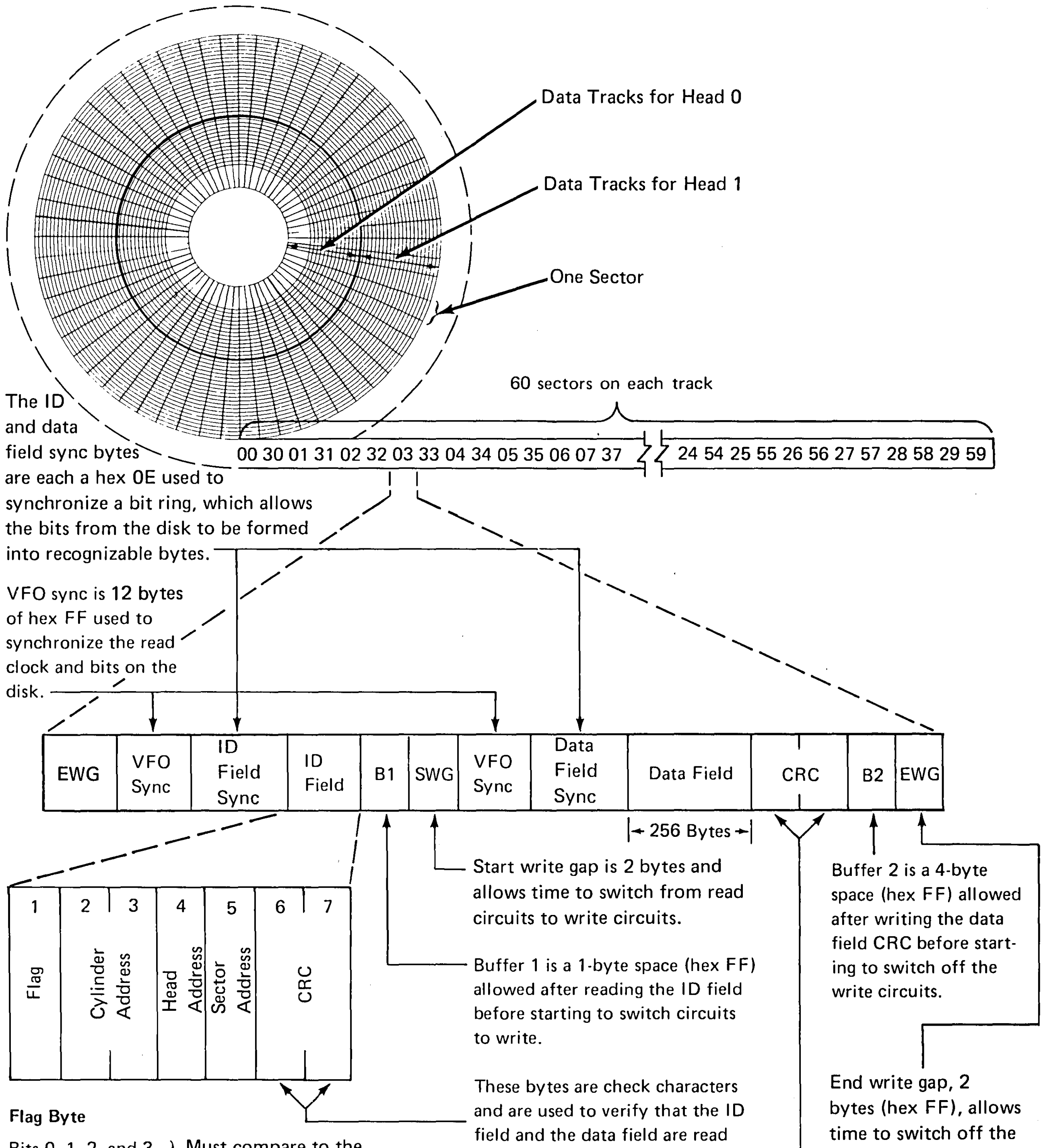


Figure 6-2. Disk Surface



The ID and data field sync bytes are each a hex 0E used to synchronize a bit ring, which allows the bits from the disk to be formed into recognizable bytes.

VFO sync is 12 bytes of hex FF used to synchronize the read clock and bits on the disk.

Start write gap is 2 bytes and allows time to switch from read circuits to write circuits.

Buffer 1 is a 1-byte space (hex FF) allowed after reading the ID field before starting to switch circuits to write.

These bytes are check characters and are used to verify that the ID field and the data field are read correctly.

Buffer 2 is a 4-byte space (hex FF) allowed after writing the data field CRC before starting to switch off the write circuits.

End write gap, 2 bytes (hex FF), allows time to switch off the write circuits and is part of the VFO sync field of the next sector.

Flag Byte

Bits 0, 1, 2, and 3 Bit 4	Must compare to the control field.
Bit 5	Must compare to the control field. If bit 5=1, data field may contain bad data.
Bits 6 and 7	00 – Good primary sector 10 – Defective primary sector 01 – Good alternate sector 11 – Defective alternate sector

Figure 6-3. Disk Sector Description

DISK OPERATIONS

The most common data operations are:

- *Read data:* Reads data from the disk and sends it to main storage.
- *Write data:* Writes data from main storage to the disk and then, if specified, verifies that the written data can be read. The verify is recommended where data integrity is critical.
- *Scan Read data:* Compares the data read from the disk to the compare field in main storage. If there is a scan hit (compare), the remainder of the sector is read into the data field in main storage.

Before all read, write or scan operations are executed, the disk actuator automatically seeks to the correct cylinder. Other automatic processing unit functions are error recovery and alternative sector processing.

Starting a Disk Operation

The input/output block that contains the information needed to execute the disk read and write operations is in the main storage program. See Figure 6-4.

The address of the 28-byte input/output block is in index register 1 when the main storage supervisor call instruction is executed. The control storage program starts the data operation between main storage and the disk storage.

00 *	01	02 *	03 *	04 *	05	06 *	07 *	08 *	09
Event Control Mask	Completion Code	Flag Byte 1	Command Code	Command Modifier	Reserved	Data Address		Sector Count -1	Flag Byte 2

0A	0B	0C	0D	0E	0F	10	11	12	13
Device Status Sense Byte 0	Device Status Sense Byte 1	Device Status Sense Byte 2	Device Status Sense Byte 3	Device Status Sense Byte 4	Reserved	TCB Pointer		Error Retry Count	Reserved

14	15	16 *	17 *	18 *	19	1A	1B
Reserved	Reserved	Sequential Sector Address	Sequential Sector Address	Sequential Sector Address	Last Sector Processed	Last Sector Processed	Last Sector Processed

*User Supplied Bytes

Figure 6-4 (Part 1 of 6). Disk Input/Output Block

Displacement of Leftmost Byte in Hex	IBM Program Label	Length in Bytes	Field Description
0	$IOBDECM	1	Event control mask *Bit* *Meaning When Set to 1* 0 Reserved 1 Data buffer in real storage 2-7 Reserved
1	$IOBDCMP	1	This is the completion code byte set by the system (when the I/O operation is complete) to inform the calling routine of the requested operation's status. It is the responsibility of the calling routine to check this byte before assuming the data is transferred without error. The following codes are used: *For All Operations Except Scan* X'40'—successful completion X'41'—permanent error *For Scan Operations* X'40'—scan hit X'41'—permanent error X'42'—scan not hit X'44'—scan equal hit
2	$IOBDFLG	1	*Bit* *Meaning When Set to 1* 0 Do not assign alternative sector on permanent error, and do not log error 1 Do not return on permanent error 2 IOS does not issue a message on permanent error 3 Do not log errors 4 Reserved 5 Reserved 6 Do not verify after write 7 Reserved
3	$IOBDCMD	1	The calling routine sets this byte to indicate the type of operation desired. Figure 6-4 (Part 5) shows the command codes and their meanings.
4	$IOBDMDR	1	The calling routine sets this byte to further define the type of operation requested. Figure 6-4 (Part 6) shows command modifiers and their meanings.
5	$IOBDUAD	1	Reserved
6	$IOBDDAT	2	This field must be initialized by the caller to contain the address of the leftmost byte of the data buffer. *Note:* Address must be on an 8-byte boundary.
8	$IOBDNB	1	The calling routine must set this byte to the hex value of the number of sectors, minus one, involved in the data transfer. For example, if five sectors are to be processed, this byte contains hex 04. This byte is not changed by the system.

Figure 6-4 (Part 2 of 6). Disk Input/Output Block

Displacement of Leftmost Byte in Hex	IBM Program Label	Length in Bytes	Field Description
9	$IOBDFL2	1	This byte contains bit indicators to request special handling of I/O operations. The bit settings are: *Bit* *Meaning When Set to 1* 0 Processor check on permanent disk error 1 Reserved 2 Delay alternate sector assignment 3 Reserved 4 Allow write to control store library 5 Allow write to SSP disk area 6 Allow access to alternative sectors or CE cylinder 7 Reserved
A	$IOBDSB0	1	These 6 bytes are used by the system to store device status information. This field need not be initialized. These bytes are not filled unless an error occurs during processing.
B	$IOBDSB1	1	
C	$IOBDSB2	1	
D	$IOBDSB3	1	
E	$IOBDSB4	1	
F	$IOBDSB5	1	
10	$IOBDTCB	2	These 2 bytes contain the address of the task control block associated with this input/output block. There is one task control block for each task that can be performed by the system. More than one task can be executing at the same time on the system. The number of tasks that can be run on a system depends on the system configuration.

Figure 6-4 (Part 3 of 6). Disk Input/Output Block

Displacement of Leftmost Byte in Hex	IBM Program Label	Length in Bytes	Field Description
12	$IOBDERR	1	Error retry count The system uses this byte to count the retries required to complete an I/O operation. This field need not be initialized.
13	$IOBDRSV	1	Reserved (should be zero)
14	$IOBDRS2	1	
15	$IOBDRS3	1	
16	$IOBDSS	3	The sequential sector number from the beginning of the disk, starting at sector 1. The caller must initialize this field before requesting the I/O operation.
19	$IOBDLSP	3	This field is modified by the system when an I/O operation is completed. It will contain the sequential sector number of the last sector which was operated upon.

Figure 6-4 (Part 4 of 6). Disk Input/Output Block

Operation		Command Code Byte[1]	Command Modifier Byte[2]
		0 1 2 3 4 5 6 7	0 1 2 3 4 5 6 7
Read	Data	1 0 1 0 0 0 0 1	0 0 0 0 0 0 0 0
	ID	1 0 1 0 0 0 0 1	0 0 0 0 0 0 0 1
	Diagnostic	1 0 1 0 0 0 0 1	0 0 0 0 0 0 1 0
	Verify	1 0 1 0 0 0 0 1	0 0 0 0 0 0 1 1
Write	Data	1 0 1 0 0 0 1 0	0 0 0 0 0 0 0 0
Scan Read	Equal	1 0 1 0 0 0 1 1	0 0 0 0 0 0 0 0
	Low or equal	1 0 1 0 0 0 1 1	0 0 0 0 0 0 0 1
	High or equal	1 0 1 0 0 0 1 1	0 0 0 0 0 0 1 0

[1] See Figure 6-4 (Part 2), byte 3 of the disk IOB.

[2] The command modifier byte specifies the function modifiers. The significance of each bit is shown in Figure 6-4 (Part 6).

Figure 6-4 (Part 5 of 6). Disk Input/Output Block

Bit	Description of Command Modifier Bits[1]
0	Data field wrap control causes the same 256-byte data area to be used for each sector accessed. Therefore, all sectors use the same data area.
1	Should be zero.
2	Should be zero.
3	Should be zero.
4	Should be zero.
5	Should be zero.
6-7	These 2 bits further define the disk I/O operation specified in the command code.

[1] See Figure 6-4 (Part 2) byte 4 and Figure 6-4 (Part 5) of the disk IOB.

Figure 6-4 (Part 6 of 6). Disk Input/Output Block

Disk Addressing

The type of addressing used by the main storage program to identify disk data areas is sequential sector addressing. The sequential sector is a binary number, starting at hexadecimal 000001 (cylinder 0, head 0, record 0) and is stepped by 1 for each sector processed. The sequential sector addressing extends through the last data sector on the first disk and continues on to the second disk if there are two disks installed. The sequential sector address is specified in the bytes at hexadecimal 16, 17, and 18 of the disk input/output block.

The sequential sector address can be calculated from the actual sector address by using the following:

For 8.6-megabyte disk systems and 13.2-megabyte disk systems:

$$\text{Sequential Sector} = 180C + 60H + S + 1$$

For 27.1-megabyte disk systems:

$$\text{Sequential Sector} = 180C + 60H + S + 1 + 54{,}240D$$

where

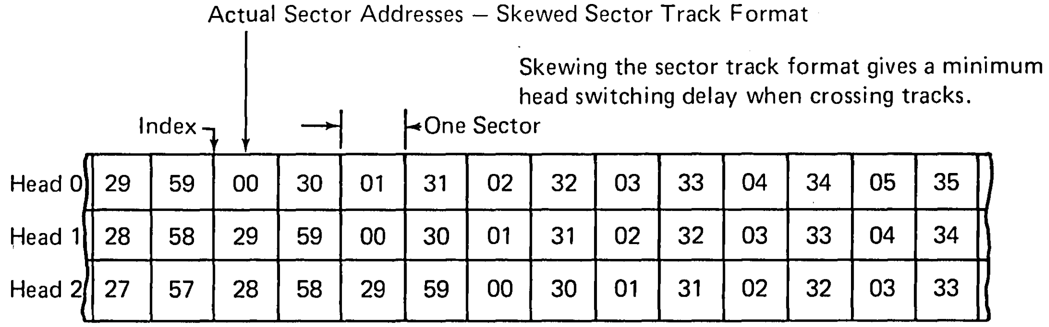
C=cylinder address

H=head address

S=actual sector address as written in the identification field, and shown in Figure 6-5.

D=0, first disk (sequential sector < or = 00D3E0 hexadecimal)

D=1, second disk (sequential sector > 00D3E0 hexadecimal)


Actual Sector Addresses – Skewed Sector Track Format
Skewing the sector track format gives a minimum head switching delay when crossing tracks.
Index
One Sector

Head 0	29	59	00	30	01	31	02	32	03	33	04	34	05	35
Head 1	28	58	29	59	00	30	01	31	02	32	03	33	04	34
Head 2	27	57	28	58	29	59	00	30	01	31	02	32	03	33

Figure 6-5. Disk Addressing

Time Needed for a Disk Operation

The time needed to execute an operation is made up of the following:

- Processing unit time needed to set up the operations.
- Seek time to the correct cylinder.
- Disk turning delay.
- Data move time.
- Head switching time, if a many sector operation is needed.
- Seek to the next cylinder, if needed on a many sector operation.
- If the operation uses both disks of a two-disk system, the above steps are taken one more time.

The seek times for the 8.6-megabyte disk are:

Cylinder to cylinder	10 ms (maximum)
Average seek (67 cylinders)	35 ms (maximum)
Maximum seek (201 cylinders)	55 ms (maximum)

The seek times for the 13.2-megabyte disk are:

Cylinder to cylinder	10 ms (maximum)
Average seek (101 cylinders)	40 ms (maximum)
Maximum seek (302 cylinders)	70 ms (maximum)

For determining the seek time for the disk use the following:

If $N \leq 100$, $T = 5.12 + 3.47(N)^{1/2} + (.02\ (100-N))^{1/2}$

If $100 < N \leq 150$, $T = 5.12 + 3.74\ (N)^{1/2}$

If $N > 150$, $T = .147N + 25.52$

where:

N = number of cylinders passed

T = maximum seek time, in milliseconds

Read Operations

Read Data

This operation automatically seeks to the correct cylinder and reads data, starting at the sector specified by the sequential sector address. The processing unit reads N+1 (N is the main sector count in the input/output block) sectors into continuous positions of main storage, starting at the address specified by the input/output block. Head switching or seeking to the next cylinder, if needed, is automatic.

Read Identification

The read identification operation seeks to the correct cylinder and reads the 5-byte identification field, from the sector specified by the sequential sector address in the input/output block.

The processing unit starts the read identification operation at the index mark and counts the sector marks to locate the specified sector. The sector identification field is read and sent to the main storage data area specified by the data address in the input/output block.

Read Data Diagnostic

This operation seeks to the correct cylinder, locates the specified sequential sector, and sends the sector data field to main storage. The processing unit starts the read data diagnostic operation at the index mark and counts sequential sector marks until the sector specified in the sequential sector field of the input/output block is located. The identification field is compared and the data field is read and sent to the main storage data area specified by the data in the input/output block. If the identification field does not compare or if a check occurs as the identification field is being read, the data field is still read and sent to main storage. The identification orientation correct bit is off if a check occurs as the identification field is being read.

Read Verify

This operation seeks to the correct cylinder and verifies that the specified sectors can be read. The processing unit verifies N+1 (N is the sector count in the input/output block) sectors starting at the specified sequential sector. Head switching or seeking to the next cylinder, if needed, is automatic.

Scan Read Equal

This operation starts at the sector specified by the sequential sector address and compares N+1 (N is the sector count in the input/output block) sectors read from disk to a single 258-byte main storage data field addressed by the data address in the input/output block. The 258-byte main storage data field is made up of compare fields followed by scan mask fields. The scan mask fields are hexadecimal 'FF' bytes and the compare fields are data which is compared byte to byte to the data read from the disk.

At the end of each compare field a test is made to determine if the data read from disk was equal to the compare field in main storage. If the two fields are not equal, scanning starts again at the next compare field. If after scanning N+1 sectors a scan hit does not occur, a completion code of hexadecimal 42 is set in byte 1 of the input/output block. If the two fields are equal, scanning stops. Starting at the end of the compare field on which scan equal hit occurs, the remainder of the 256-byte sector is read into the data field in main storage, the operation is ended, and a completion code of hexadecimal 44 is set in byte 1 of the input/output block. The data that is read into main storage is moved 2 bytes to a higher address. Therefore, the 2 bytes following the compare field in main storage are not changed.

The following is a guide to the format of the main storage data field.

1. The compare and scan mask fields can be 1 or more bytes long.

2. The 258-byte main storage field may start with a scan mask field.

3. Each compare field must be followed by a scan mask field.

4. The last scan mask field must extend to the end of the 258-byte data field and be at least 2 bytes long.

5. If the main storage data field is 258 bytes of hexadecimal FF, the completion code will be hexadecimal 42. If the last compare field in the 258-byte main storage data field ends with byte 256, the completion code will indicate if a scan hit occurred, but no data will be read into main storage. Head switching or seeking to the next cylinder, if needed, is automatic.

Examples of the starting and ending status of the main storage data field for a scan read command are shown in Figure 6-6.

Example 1: A scan hit occurs on the first compare field of the third sector scanned.

Contents of the main storage data field at start of the 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 the third sector scanned by the 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 the main storage data field at end of the operation:

0	1	4	3	FF	J	ƀ	1	8	.	0	0	ƀ	0	2	G	R	#	5	7	2	ƀ	S	O	C

Completion code = X'44'.

Example 2: A scan hit occurs on the second compare field.

Contents of the main storage data field at start of the 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 the sector 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 the main storage data field at end of the operation:

0	1	4	3	FF	J	G	J	O	N	E	S	FF	FF	ƀ	ƀ	ƀ	ƀ	ƀ	1

Completion code = X'44'.

Figure 6-6. Main Storage Starting and Ending Status

Scan Read Low or Equal

The scan read low or equal operation is the same as the scan read equal operation except that a scan hit decision occurs if the disk data is a binary value which is less than or equal to the compare field in main storage. At the end of each compare field a test is made to determine if the data read from the disk was less than or equal to the compare field in main storage. If the disk data was more than the compare field, scanning starts again with the next compare field. If after scanning N+1 sectors a scan hit does not occur, a completion code of hexadecimal 42 is set in the input/output block and no data is sent to main storage. If the disk data was less than the compare field, the remainder of the 256-byte sector is read (with no more compare) and sent to main storage moved by a two byte higher main storage address, and a completion code of hexadecimal 40 is set in the input/output block. If the two compare fields were equal then the remainder of the 256-byte sector is read into main storage and moved 2 bytes to a higher address, and a completion code of hexadecimal 44 is set in the input/output block.

Scan Read High or Equal

The scan read high or equal operation is the same as scan read low or equal operation except that the scan hit decision occurs if the disk data is a binary value which is larger than or equal to the compare field in main storage.

Write Operations

Write Data

This operation automatically seeks to the correct cylinder, locates the specified sequential sector and sends data from the main storage data area specified by the data address in the input/output block to the disk. This operation starts at the sector specified by the sequential sector address and N+1 (N is the sector count in the input/output block) sectors are written. Head switching and seeking to the next cylinder, if needed, is automatic.

The disk attachment generates 2 cyclic redundancy check bytes for each data field. These 2 bytes are written in the cyclic redundancy check field on the disk.

CHECK CONDITIONS AND STATUS

When an error occurs, the device status bytes show the conditions that result after executing or attempting to execute an operation requested by the input/output block. The processing unit automatically executes disk error recover routines.

Status Byte 0

Bit Description

0 *Disk not ready:* Indicates that the disk (1) did not come ready after the initial power up sequence, (2) is not turning at the correct speed, (3) is unsafe, or (4) has a motor brake failure.

1 *Alternative sector process:* Indicates the error recovery failed while attempting to process an alternative sector. This bit is set on during the first seek from the primary track to the alternative track and is reset when the seek back to the primary track is complete. A disk read, write, or scan operation resets this bit.

2 *Sector sync check:* Indicates an error was found while reading the identification sync byte or the data field sync byte. A disk read, write, or scan operation resets this bit.

3 *Off track check:* Indicates that the actuator arm moved off the servo track when not seeking. A disk read, write, or seek operation resets this bit.

4 *Cyclic redundancy check:* Indicates a cyclic redundancy check occurred on either an identification field or a data field. The no record found status is set at the same time. A disk read, write, or scan operation resets this bit.

5 *Parallel data bus out parity check:* Indicates a parity error on the data bus out as data is sent from main storage to the disk. A disk read, write, or scan operation resets this bit.

6 *Write data echo check:* Indicates the serial write data and write data echo from the disk do not compare. This check can only occur during a write operation. A disk read, write, or scan operation resets this bit.

7 *Cycle steal overrun:* Indicates that the processing unit did not move data between the disk and main storage quickly enough, or it moved data too quickly and lost a byte. A disk read, write, or scan operation resets this bit.

Status Byte 1

Bit Description

0 *No operation:* Set when a disk read, write, or scan operation is used while the disk is not ready. A disk read, write, or scan operation resets this bit.

1 *Data not safe:* Indicates a select unsafe, write unsafe, or servo unsafe condition. This indicates that the hardware has failed so that errors may not be found during processing or that data may be lost. (The not ready status bit is set at the same time).

2 *Not valid seek address:* Indicates that the file attempted to seek to a cylinder address higher than any on the disk. A read, write, or scan operation resets this bit.

3 *Attachment equipment check:* Indicates a hardware check. For a list of the conditions that set equipment check, see Figure 6-5.

4 *No record found:* Indicates that the sector specified in the sequential sector field was not found in two complete turns of the disk. This normally results from a seek failure, finding an alternative sector, or surface damage in the identification field. A disk read, write, or seek operation resets this bit.

5 *Scan equal hit:* This bit is set on in the input/output block if a scan equal condition occurred during a scan read operation, and some type of an error also occurred.

6 *Scan not hit:* This bit is set on in the input/output block if a scan hit condition was not found during a scan read operation, and some type of an error also occurred.

7 *Seek Check:* Set if any adapter checks occur during a seek operation. A read, write, or scan operation resets this bit.

Status Byte 2

Bit Description

0 *Serializer/deserializer check:* Indicates that a parity check occurred in the hardware. A disk read, write, or scan operation resets this bit.

1 *Write check:* Indicates that a data head is writing on the disk when it should not be or that it is not writing when it should be. A disk read, write, or scan operation resets this bit.

2 *Channel transfer error:* Indicates that a parity failure was found as data was sent from the disk to main storage. A disk read, write, or scan operation resets this bit.

3 *Phase lock oscillator out of sync:* Indicates the phase lock oscillator is out of sync. A disk read, write, or scan operation resets this bit.

4 *Interrupt time-out check:* The disk attachment did not request an interrupt by two seconds after an interrupt was enabled.

5 *Behind home:* The actuator is behind home.

6 *Identification orientation correct:* Indicates that the correct identification orientation occurred during any read data, write data or scan read operation.

7 *Sector check:* A sector or index pulse occurred at the wrong time.

Status Byte 3

Bit Description

0 *A second disk is installed:* The second disk storage drive is installed.

1 *Select unsafe:* Indicates that more than one head is selected during a write operation. (The data unsafe and not ready status bit are set at the same time.) Correcting the error condition resets the select unsafe bit.

2 *Write unsafe:* Indicates that one of the following conditions were found during a write operation. (1) Write was selected and no writing occurred, or (2) write was not selected and write current was on. (3) Write selected with no head selected. (The data unsafe and not ready bits are set at the same time.) Correcting the error condition resets the write unsafe bit.

3 *Brake failure:* Indicates a failure in the disk drive motor brake. (The not ready status bit is set at the same time.) The system disconnects AC power from both disks. The system power must be turned off to reset this bit.

4 *Servo unsafe:* Indicates that one of the following conditions was found during a write operation. (1) Write was selected and the head was off the track, or (2) write was selected and the phase lock oscillator was out of sync. (The data unsafe and not ready bits are set at the same time.) Correcting the error condition resets the servo unsafe bit.

5 *Not used*

6 *Not used*

7 *Storage indicator:* When this bit is on, the installed disks have 13.2 megabytes of storage each. When this bit is off, the installed disk has 8.6 megabytes of storage capacity.

Status Byte 4

Bit Description

0 *Not valid input/output buffer address:* Indicates that the I/O buffer address was not in the user's area of main storage or that it did not start on an 8-byte boundary.

1-5 *Not used*

6,7 *Head select sense bits:*

Head selected	Bit 6	Bit 7
0	1	1
1	1	0
2	0	1

Status Byte 0

Bit	Description	Read Data	Read ID	Read Diagnostic	Read Verify	Write Data	Scan High or Equal	Scan Equal	Scan Low or Equal
0	*Disk not ready* is part of unit-check condition.	X	X	X	X	X	X	X	X
1	*Alternate sector process* is part of unit-check condition.	X			X	X	X	X	X
2	*Sector sync check* is part of unit-check condition. Equipment check is set concurrently.	X	X	X	X	X	X	X	X
3	*Off track check* is part of unit-check condition. Equipment check is set concurrently.	X	X	X	X	X	X	X	X
4	*CRC* is part of unit-check condition. Equipment check is set concurrently.	X	X	X	X	X	X	X	X
5	*Parallel DBO parity check* is part of unit-check condition. Equipment check is set concurrently.	X	X	X	X	X	X	X	X
6	*Write data echo check* is part of unit-check condition. Equipment check is set concurrently.					X			
7	*Cycle steal overrun* is part of unit-check condition. Equipment check is set concurrently.	X	X	X	X	X	X	X	X

Status Byte 1

Bit	Description	Read Data	Read ID	Read Diagnostic	Read Verify	Write Data	Scan High or Equal	Scan Equal	Scan Low or Equal
0	*No operation* is part of unit-check condition.	X	X	X	X	X	X	X	X
1	*Data not safe* is part of unit-check condition.	X	X	X	X	X	X	X	X
2	*Invalid seek address* is part of unit-check condition. Equipment check is set at the same time.	X	X	X	X	X	X	X	X
3	*Attachment equipment check* is part of unit-check condition.	X	X	X	X	X	X	X	X
4	*No record found* is part of unit-check condition.	X			X	X	X	X	X
5	*Scan equal hit.*						X	X	X
6	*Scan not hit.*						X	X	X
7	*Seek check* is part of unit-check condition. Equipment check is set concurrently.	X	X	X	X	X	X	X	X

Status Byte 2

Bit	Description	Read Data	Read ID	Read Diagnostic	Read Verify	Write Data	Scan Read High or Equal	Scan Read Equal	Scan Read Low or Equal
0	*Serializer-deserializer check* is part of unit-check condition.	X	X	X	X	X	X	X	X
1	*Write check* and equipment check are set concurrently.	X	X	X	X	X	X	X	X
2	*Channel transfer error* is part of unit-check condition. Equipment check is set concurrently.	X	X	X			X	X	X
3	*PLO out of sync.*	X	X	X	X	X	X	X	X
4	*Interrupt timeout check.*	X	X	X	X	X	X	X	X
5	*Behind home.*								
6	*ID orientation correct.*	X	X	X	X	X	X	X	X
7	*Sector count or data check.*	X	X	X	X	X	X	X	X

Status Byte 3

Bit	Description	Read Data	Read ID	Read Diagnostic	Read Verify	Write Data	Scan High or Equal	Scan Equal	Scan Low or Equal
0	Disk Storage Drive B is installed (27.1 Mb).	X	X	X	X	X	X	X	X
1	*Select unsafe* is part of unit-check condition.					X			
2	*Write unsafe* is part of unit-check condition.	X	X	X	X	X	X	X	X
3	*Brake failure* is part of unit-check condition.	X	X	X	X	X	X	X	X
4	*Servo unsafe* is part of unit-check condition.					X			
5	Reserved.								
6	Reserved.								
7	*Capacity indicator.* **8.6 Mb** **13.2 Mb or 27.1 Mb** 0 1	X	X	X	X	X	X	X	X

Status Byte 4

Bit	Description	Read Data	Read ID	Read Diagnostic	Read Verify	Write Data	Scan High or Equal	Scan Equal	Scan Low or Equal
0	*Not valid I/O buffer address.*	X	X	X		X	X	X	X
1	Reserved.								
2	Reserved.								
3	Reserved.								
4	Reserved.								
5	Reserved.								
6	*Head select bits:*	X	X	X	X	X	X	X	X
7		X	X	X	X	X	X	X	X

Head 0	Head 1	Head 2
1	1	0
1	0	1

DISK OPERATING PROCEDURES

Disk Program Load

Pressing the Load key on the operator panel starts the program load. The CSIPL and MSIPL switches on the CE panel determine which device reads data during the control storage initial program load and the main storage initial program load.

A disk initial control storage program load procedure starts an immediate recalibrate to cylinder 000, track 0 and loads sectors 00 through 15 (2048 words) into control storage locations from address 0000 through 2047. The processing unit starts main storage initial program load from the disk during the execution of control storage initial program load if the MSIPL switch is set to Disk. The Load light turns off when the processing unit completes a correct main storage initial program load. The display screen signals the end of the initial program load if the IBM system support programming is used.

Alternative Sector Assignment

The processing unit uses an alternative sector assignment procedure for assigning alternative sectors to damaged disk sectors. If a sector is damaged the processing unit automatically assigns a good alternative sector. The processing unit does not use the procedure for cylinders 000 and 001 on the first disk, which must not be damaged, nor for alternate tracks and CE cylinders on either disk.

In the alternative assignment procedure, the processing unit:

1. Recovers the data part from the damaged sector.

2. Seeks to the alternative sector cylinder (see Figure 6-2) and locates the next available good alternative sector, using a read identification operation.

3. Writes the data recovered from the damaged sector in the data field of the alternative sector.

4. Writes hexadecimal 01 in the flag byte of the alternative sector to indicate a good alternative sector and writes the address of the damaged sector in the remainder of the identification field. Then the processing unit verifies the identification field.

5. Seeks to the primary track and changes the identification field of the damaged sector, writing hexadecimal 02 in the flag byte of the identification field and writing the address of the alternative sector in the remainder of the identification field.

6. Verifies the identification field. If the identification field cannot be read, the identification field is written using a write identification with skew operation. Then the processing unit verifies the identification field using a read identification with skew operation.

Alternative Sector Processing

If the identification field does not compare during read data, read verify, write data, or scan read operation, the processing unit automatically reads the identification field. If the identification field has an alternative sector assigned, the processing unit automatically seeks to an alternative sector and executes the specified operation on the assigned alternative sector.

If the identification field does not compare on the alternative sector, the processing unit sets the no record found status bit and alternative sector processing status bit in the disk status byte. If the identification field does compare on the alternative sector, the assigned alternative sector is processed. After processing the assigned alternative sector, the processing unit seeks back to the primary track and processing continues.

Error Recovery

The processing unit executes disk error recovery if bit 0 of the input/output block flag byte is set to 0. If bit 0 of the input/output block flag byte is set to 1, the processing unit does not do error recovery. Check conditions are used for error recovery in the following way:

Seek Check, Not Valid Seek Address Check, or No Record Found: The processing unit resets the check, recalibrates, and seeks to the correct cylinder.

Data Unsafe or Not Ready: The first time this check occurs, the data unsafe condition is reset, a recalibrate and seek to the current cylinder is executed, and the operation is executed again. The second and all following operations are not executed and the processing unit sets the no operation status bit. The processing unit must be loaded again to recover from the no operation status because of a data unsafe condition.

All Other Checks:. The processing unit resets the check and starts the operation over again.

Error Recovery by the IBM Input/Output Supervisor Program

If a unit check continues for 17 attempts to execute the operation, the type of unit check indicates which type of error recovery will be used. If the processing unit determines that the error cannot be recovered from, the processing unit error routine assembles an operator message which indicates the error cannot be recovered from. If the error can be recovered from, the processing unit also enters an alternative sector assignment routine. For all unit check status conditions, the processing unit error routine logs the error data.

Chapter 7. Work Stations

Two types of work stations can be attached to the IBM System/34: the 5251 Display Station and the 5256 Printer. Because of programming characteristics, the 5256 Printer is described with the 5211 Printer in Chapter 5, *Printer Functions*.

System/34 supports up to eight work stations. A 5251 Display Station *must* be attached to each System/34 as the system console. It is the *only* work station attached to cable entry position 0 on the cable entry tower. All other work stations are optional, and must be attached to cable entry positions 1 through 3. The cable-thru feature is used to attach more than one work station to these three cable entry positions.

System/34 supplies the commands, orders, and control characters needed to direct the operation of the work stations. The display station commands control such operations as writing, reading, erasing of data, and cursor positioning; they also control the indicators on the display station. Printer commands control printing, formatting, and forms movement. The commands and the input/output blocks, status bytes, and error recovery procedures for the display station are described later in this chapter. For the 5256 Printer, they are described in Chapter 5.

PHYSICAL CHARACTERISTICS OF THE 5251 DISPLAY STATION

The 5251 Display Station is a fully buffered display station used for inquiry and data entry applications. The major parts of the 5251 are the display screen and keyboard.

The display screen can display up to 1920 characters at the same time. The characters are displayed in 24 rows with 80 characters in each row. A marker (called a cursor) indicates the position on the screen where the next character will be entered. The cursor can be moved to any position on the screen either by the operator (by pressing the cursor motion keys), or under control of the system program.

The keyboard, shown in Figure 7-1, supports the character set shown in Figure 7-2. For a description of the keys, see *Keyboard Key Functions* later in this chapter.

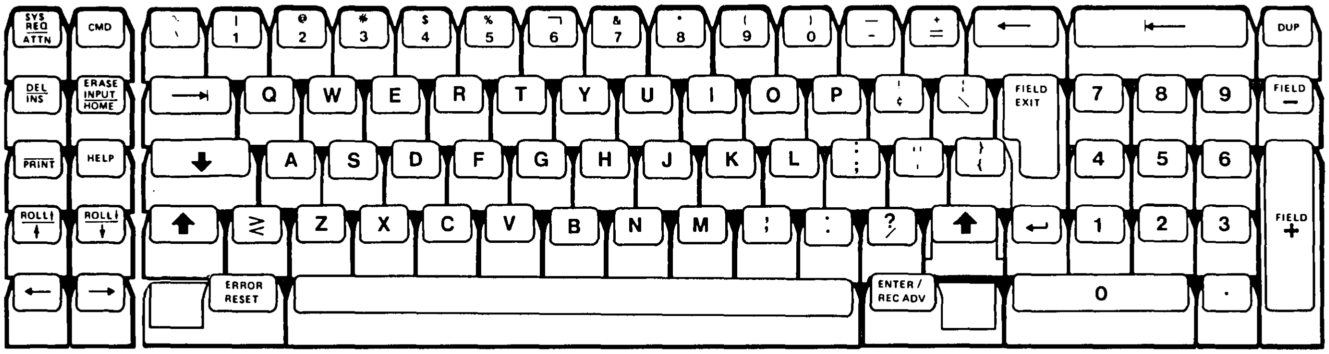


Figure 7-1. 5251 Keyboard (United States)

2nd Hex Char ↓ 1st Hex Char ←	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	(see note)		Used for Display Attributes		sp	&	-						{	}	\	0
1							/		a	j	~		A	J		1
2									b	k	s		B	K	S	2
3									c	l	t		C	L	T	3
4									d	m	u		D	M	U	4
5									e	n	v		E	N	V	5
6									f	o	w		F	O	W	6
7									g	p	x		G	P	X	7
8									h	q	y		H	Q	Y	8
9								`	i	r	z		I	R	Z	9
A					¢	!	¦	:								
B					.	$	,	#								
C		$\overline{*}$			<	*	%	@								
D					(	)	_	'								
E					+	;	>	=								
F					|	¬	?	"								

Note: The null character is hex 00.

Figure 7-2. Character Set for Display Station

OPERATIONAL CHARACTERISTICS OF THE 5251 DISPLAY STATION

The operational characteristics for the 5251 Display Station include operator aids, keyboard key functions and display station modes.

Operator Aids

The display station has aids that inform the operator of specific operating conditions. These aids include 10 indicators, (five program status indicators and five hardware status indicators), a clicker, an audible alarm, and a set of four-digit error codes.

Indicators

The five program status indicators (lighted squares on the right-hand side of the display screen) and the function of each are:

System Available: Indicates that the work station controller in System/34 is polling the display station; data can be entered from the keyboard.

Message Waiting: Indicates that a message for this display station is in the system. The operator must press the Sys Req/Attn key or press (and hold) a shift key followed by the Sys Req/Attn key and the Enter/Rec Adv key to display the message.

Keyboard Shift: Indicates that the keyboard is in upper-shift.

Insert Mode: Indicates that the display station is in insert mode; characters can be inserted into a data field.

Input Inhibited: Indicates that System/34 will not accept input from the keyboard with the exception of the shift keys and the Sys Req/Attn key. However, if the display station is in operator-error mode when this indicator is on, the Help key and Error Reset key are also accepted.

The five hardware status indicators and the function of each are:

Line Sync: Indicates that error-free data is being received.

Line Check: Indicates that a command byte or data byte with wrong parity has been received from System/34.

Internal Check: Indicates that a display station parity error has occurred.

Storage Check: Indicates that a display station parity error has occurred on the extended storage card.

Ready Indicator: Indicates the status of the display station; either ready or not ready.

Clicker

When the operator is entering information into the system through the keyboard, a *clicking* sound occurs each time a key is pressed. However, if the keyboard is locked, the clicker is inhibited and no sound occurs.

Audible Alarm

An audible alarm on the keyboard is activated by the system program when the display station needs operator attention. Once activated, the alarm will sound for approximately 1 second.

Four-Digit Error Codes

When an operator presses the wrong key and the keying error is sensed by the work station controller:

1. The display station is forced into operator-error mode (see *Operator-Error Mode* later in this chapter).
2. The keyboard locks (clicker disabled) and the Input Inhibited indicator comes on. Only the Error Reset, Help, Attn, and shift keys are operational when the keyboard is locked.
3. The cursor blinks and is placed under the character position in error, or under the first position of the field in error.
4. A four-digit error code is displayed on the bottom row (left side) of the screen; the four-digit error code blinks and is intensified. The four-digit error code replaces the characters, if any, that were on the bottom row.

If the Error Reset key is pressed, the four-digit error code is cleared and the characters that were on the bottom row are again displayed. Pressing the Error Reset key also unlocks the keyboard if the keyboard was not locked before the error was sensed.

If the Help key is pressed, an AID code of hex FB and the four-digit error code are sent to the host. The host responds with a message that clears the four-digit error code, and a message number and description of the operator error is displayed on the bottom row of the screen. The message number is preceded by the letters KBD. To recover from any of the operator errors, see the KBD messages in the *Displayed Messages Guide.*

Keyboard Key Functions

The keyboard for the 5251 Display Station includes three different groups of keys. The alphameric keys are similar to the keys on a typewriter (letters, numbers, and special characters); they are used to enter alphabetic and numeric data. The numeric keys on the right side of the keyboard are used for entering numeric data only. The remaining keys are divided into:

- Signal keys
- AID request keys
- Cursor motion keys
- Editing keys
- Special field keys

The signal keys and the AID request keys are AID generating keys; that is, when one of these keys (or a combination of these keys) is pressed, an AID byte is stored in the work station controller. How the AID bytes are sent to the host, the priorities of the AID bytes, and other information is given under *Signal Keys* and *AID Request Keys*. All of the AID bytes are shown in Figure 7-3.

Depending on the state of the keyboard (locked or unlocked), the signal keys may or may not be operational, but none of the AID request keys are operational when the keyboard is locked. When these keys are operational is shown in Figure 7-4. Figure 7-4 also shows the relationship of these keys to the mandatory fields (fill and enter), to the check digit verification fields, and to insert mode.

Key or Function	AID Code (Note 1)	Key or Function	AID Code
Command 1	31	Printer Cancel (Note 2)	D1
2	32		
3	33	Printer Operation Complete (Note 2)	D3
4	34	Printer Transfer Complete (Note 2)	D4
5	35		
6	36	Sys Req/Enter (2 keys)	F0
7	37	Enter/Rec Adv	F1
8	38	Attention	F2
9	39	Help (not in operator-error mode)	F3
10	3A	Roll Down	F4
11	3B	Roll Up	F5
12	3C	Print	F6
Test Request	3D	Record Backspace	F8
		Resources Available (Note 3)	FA
Command 13	B1	Help (in operator-error mode)	FB
14	B2	Work Station Controller Sensed Error AID (Note 2)	FF
15	B3		
16	B4		
17	B5		
18	B6		
19	B7		
20	B8		
21	B9		
22	BA		
23	BB		
24	BC	Work Station Input/Output Sensed Error AID (Note 2)	04
Clear	BD		

Notes:

1. The AID codes in this table are given in hexadecimal notation; they can be found in byte 13 of the input/output block (IOB).
2. The AID is not generated from the keyboard.
3. The AID cannot be generated under control of the keyboard operator even though the Error Reset key must be pressed.

Figure 7-3. Attention Identifier (AID) Bytes

Signal Keys

The AID bytes generated by the signal keys (or by a signal key in combination with an AID request key) are sent to the host under control of the enable-interrupt bit (bit 2 of the device control byte). The device control byte is in the work station control field, and is described in the *Data Areas Handbook*.

The work station input/output control handler (WSIOCH) microcode sets the enable-interrupt bit on to allow the signal key AIDS to be sent to the host. Although the enable-interrupt bit must be on before the AID byte can be sent to the host, the invite-response-allowed bit (bit 0 of the same device control byte) can be either on or off. In other words, the user cannot inhibit the AID byte from being sent to the host.

Other characteristics of the AID bytes generated by the signal keys are:

- They are of a higher priority than the AID bytes generated by the AID request keys.

- More than 1 can be stored in the work station controller, but only 1 is sent at a time to the host. They are sent in the following priority:
  a. FA (Resources Available)
  b. FB (Help key in operator-error mode)
  c. F2 (Attn)
  d. F0 (Sys Req/Enter)

The signal keys include the Sys Req/Attn key, the Error Reset key (in system request mode or operator-error mode), the Help key (in operator-error mode), and the Enter/Rec Adv key (Enter, in system request mode); all are described in the following paragraphs.

Sys Req: Forces the display station into system request mode if (1) the Sys Req/Attn key is pressed while a shift key is held down, and (2) the display station is not in command key mode, insert mode, or operator-error mode. Once the display station is in system request mode, the only operational keys are Enter/Rec Adv, Error Reset, and shift. Pressing any other key causes an operator error.

No AID byte is generated when the Sys Req/Attn key is pressed (while a shift key is held down). Instead, the AID byte is generated after the display station is in system request mode, and the Enter/Rec Adv key or the Error Reset key is pressed.

The system request function of the Sys Req/Attn key is used in free key mode to sign on to the system or to get messages displayed that are waiting in the system (indicated by the Message Waiting indicator). A three-key sequence is needed to sign on to the system or to get a message displayed; press and hold a shift key, press the Sys Req/Attn key, and then press the Enter/Rec Adv key.

Attn: Activated by pressing the Sys Req/Attn key if (1) a shift key is not pressed at the same time, and (2) the display station is not in free key mode, command key mode, or system request mode.

An AID byte of hex F2 is generated when the Sys Req/Attn key is pressed (and a shift key is not pressed at the same time). The AID byte is generated if the keyboard is locked or unlocked, and the state of the keyboard (locked or unlocked) does not change.

The attention function of the Sys Req/Attn key is used to do an inquiry function. By pressing the Sys Req/Attn key, the system causes an option menu to be displayed on the screen.

Error Reset (System Request Mode or Operator-Error Mode): Functions according to the mode the display station is in when the Error Reset key is pressed. If the display station is in system request mode, this key causes:

- A reset of system request mode.

- The keyboard to lock; the same state the keyboard was in before pressing the Sys Req/Attn key.

- An AID byte of hex FA if a command was rejected during system request mode. If a command was not rejected, no AID byte is generated.

If the display station is in operator-error mode, this key causes:

- A reset of operator-error mode.

- The keyboard to the same state (locked or unlocked) the keyboard was in before pressing the Sys Req/Attn key.

- The display station to restore the last line of data.

- An AID byte of hex FA if a command was rejected during operator-error mode. If a command was not rejected, no AID byte is generated.

Key or Function	Insert Mode	Check Digit and/or Mandatory Fill Field	Mandatory Enter Field	Keyboard Lock State
Commands 1-24	A	Y	Y	U
Test Request	A	N	N	U
Clear	A	N	N	U
Attention	B	N	N	L
System Request/Enter	E	Y	N	L
Print	E	Y	N	U
Roll Up	E	Y	Y	U
Roll Down	E	Y	Y	U
Record Backspace	E	N	N	U
Enter/Rec Adv	R	Y	Y	U
Help (not operator-error mode)	R	N	N	U
Help (operator-error mode)	A	N	N	L

Legend

Insert Mode

A = Key or function is not operational if display station is in insert mode.
B = Key or function is operational if display station is in insert mode, or if it is not in insert mode.
E = Error; four-digit code is 0017.
R = Insert mode is reset when key is pressed.

Check Digit, Mandatory Fill, and Mandatory Enter Fields

Y = Yes; these fields are checked.
N = No; these fields are not checked.

Keyboard Lock State

U = Unlocked; the key or function is not operational unless the keyboard is unlocked.
L = Locked; the key or function is operational even though the keyboard is locked. However, the System Request/Enter function is not operational in operator-error mode, and Help (operator-error mode) is operational only when the keyboard is locked and the display station is in operator-error mode.

Figure 7-4. Signal Keys and AID Request Keys

See *Error Reset* later in this chapter for more information on the Error Reset key.

Help (Operator-Error Mode): Generates a four-digit error code, and an AID byte of hex FB. The error code is sent to the host with the AID byte. The Help key (when pressed in operator-error mode) also locks the keyboard. Then, the only keys that are valid are the Attn key and the shift keys. Because the Error Reset key is not valid, the operator cannot unlock the keyboard. To enable the operator to unlock the keyboard, the SSP must issue a write error command.

Issuing a write error command helps the operator during error recovery because an error message, taken from the disk, is sent to the appropriate display station where it is displayed on the bottom row of the display screen. The displayed message gives the operator more information on the type of error so that the operator can recover from the error.

Enter (System Request Mode): Resets system request mode, restores the keyboard to the locked state (same state the keyboard was in before pressing the Sys Req/Attn key), and causes an AID byte of F0 to be posted in the work station controller. If an error was sensed in system request mode and the Enter/Rec Adv key is pressed, the display station is forced into operator-error mode.

The display station is also forced into operator-error mode if the Enter/Rec Adv key is pressed when the cursor is in a mandatory fill field or a check digit field, and the requirements of these fields are not satisfied. For example, the mandatory fill field is not filled when the Enter/Rec Adv key is pressed. If the requirements of these fields are satisfied, the Enter/Rec Adv key causes an AID byte of hex F0 to be posted in the work station controller.

When the Enter/Rec Adv key is pressed, an AID byte of hex FA is also posted in the work station controller if a command was rejected during system request mode. The AID byte of hex FA has priority over the AID byte of hex F0.

See *Enter/Rec Adv* later in this chapter for more information on the Enter/Rec Adv key.

AID Request Keys

The AID bytes generated by the AID request keys are sent to the host under control of the user if the invite-response-allowed bit (bit 0 of the device control byte) and the enable-interrupt bit (bit 2 of the device control byte) are on. The device control byte is in the work station control field, and is described in the *Data Areas Handbook.*

Other characteristics of the AID bytes generated by the AID request keys, and of the AID request keys, are:

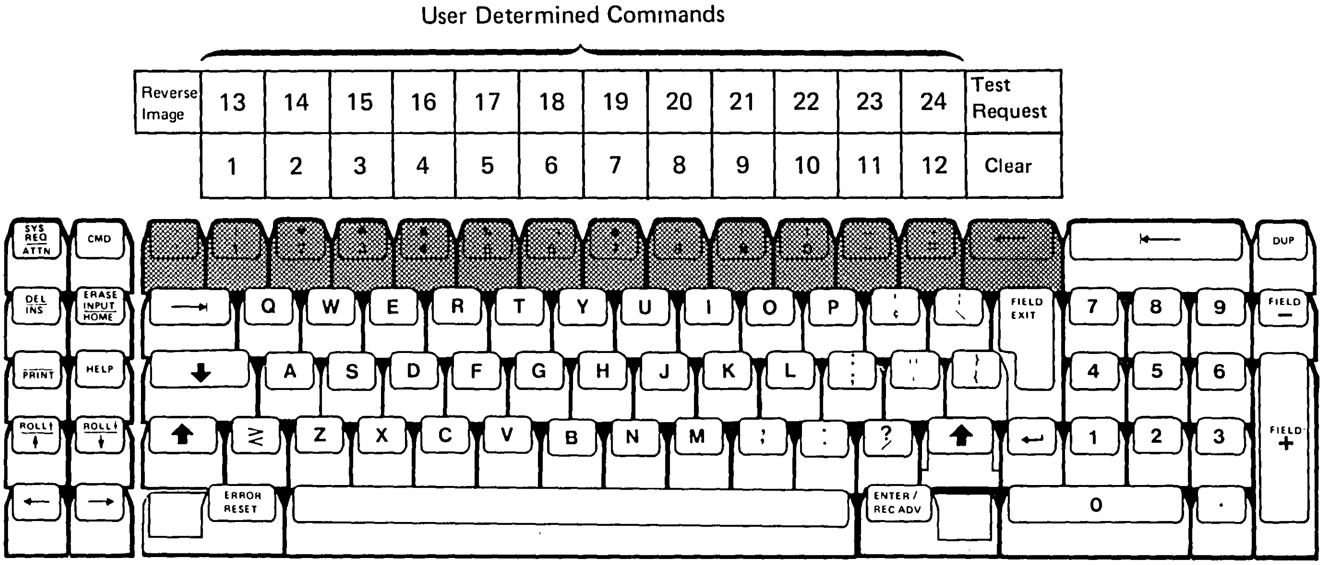
- None of these keys are operational if the keyboard is locked, if the display station is in free key mode, or if the cursor is in an active, right-adjust field.
- All of these keys force the keyboard into a locked state, and an AID byte to be stored in the work station controller.
- The AID bytes are of a lower priority than the AID bytes generated by the signal keys.

The AID request keys include the Cmd key (followed by a valid command key), the Enter/Rec Adv key, the Help key, the Print key, and the Roll keys. All of these keys are described in the following paragraphs.

Cmd: Puts the display station into command key mode if the display station is not in operator-error mode, in system request mode, or in insert mode.

No AID byte is generated when the Cmd key is pressed. Instead, the AID byte is generated after the display station is in command key mode, and a valid command key is pressed. However, the AID byte is not sent to the host until all field checks (mandatory fill, mandatory enter, and so on) are performed. The valid command keys are shown in Figure 7-5, and the AID bytes for these keys are shown in Figure 7-3.

The Cmd key, used in combination with the valid command keys, assigns the clear, test request, or reverse image function, or assigns 1 of 24 commands as determined by the user's program. The 24 commands determined by the user's program should be marked on a template located above the keyboard. To perform a command assigned to the bottom row of the template (commands 1 through 12), press the Cmd key and a valid command key. To perform a command assigned to the top row of the template (commands 13 through 24), press the Cmd key, press and hold a shift key, and then press a valid command key.



Note: If any key other than a shift key, Error Reset key, or valid command key is pressed after the Cmd key is pressed, operator error 0003 is displayed. If the Error Reset key is pressed, command key mode is reset.

Figure 7-5. Valid Command Keys

Enter/Rec Adv: Informs the system that the operator has completed entering on the current screen. If there is no keying error before the Enter/Rec Adv key is pressed, an AID byte of hex F1 is generated.

For information on the Enter/Rec Adv key when the display station is in system request mode, see *Enter (System Request Mode)* earlier in this chapter.

Help: Causes an AID byte of hex F3 to be sent to the host without a four-digit error code. For information on the Help key when the display station is in operator-error mode, see *Help (Operator-Error Mode)* earlier in this chapter.

Print: Informs the host with an AID byte of hex F6 that the operator wants to print the contents of the current display.

Roll (Up or Down): Informs the host that the operator wants to roll the information on the display screen up (AID byte of hex F5), or down (AID byte of hex F4).

To roll the information on the display screen up or down, press and hold a shift key and then press the appropriate roll key. All rolls are defined by, and under control of, the host.

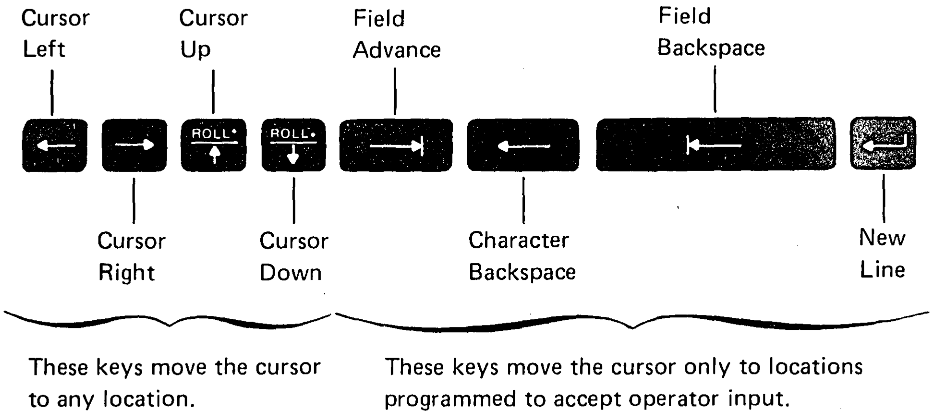

Cursor Left
Cursor Up
Field Advance
Field Backspace
ROLL•
ROLL•
Cursor Right
Cursor Down
Character Backspace
New Line
These keys move the cursor to any location.
These keys move the cursor only to locations programmed to accept operator input.

Figure 7-6. Cursor Motion Keys

Cursor Motion Keys

The cursor motion keys, as shown in Figure 7-6, permit positioning of the cursor to any character position on the display screen. These keys cause the cursor to move in the direction indicated by the arrows on the key. None of the cursor motion keys are operational when the keyboard is locked.

If any cursor motion key causes the cursor to leave an automatic enter field or a right adjust field, the automatic enter or right adjust functions are not executed even though the function is selected in the field format word. Also, there are limitations on the use of these keys when the cursor is in a mandatory fill field (see *Mandatory Fill (Bits 13-15)* later in this chapter).

Cursor Up, Down, Left, and Right: Moves the cursor either one line (up or down) or one character position (left or right) in the direction indicated by the arrow on the key. These keys also cause the cursor to wrap to the next line (cursor right), to the previous line (cursor left), to the bottom line (cursor up), or to the top line (cursor down).

If the cursor is blinking at the end of an input field (indicates that the operator must press one of the field-type keys) and the Cursor Left key is pressed, the cursor stops blinking but does not move. If the key is pressed a second time, the cursor moves one character position to the left.

Field Advance: Moves the cursor to the first position of the next unprotected field on the screen. If there are no unprotected fields on the screen, the cursor moves to the position specified by the home address. If the home address is not specified, the cursor automatically moves to the first position of row 1.

Character Backspace: Functions the same as the Cursor Left key unless the cursor is in the first position of a field. If the cursor is in the first position of a field, the cursor moves to the left to the last position of the preceding unprotected field. If the new field is a signed numeric field, the cursor moves to the position before the sign position.

If the cursor is blinking at the end of an input field (indicates that the operator must press one of the field-type keys) and the Character Backspace key is pressed, the cursor stops blinking but does not move. If the key is pressed a second time, the cursor moves one character position to the left.

Field Backspace: If the cursor is in any position of an unprotected field other than the first position, the cursor moves to the first position of the same field. If the cursor is in the first position of an unprotected field or in any position of a protected field, the cursor moves to the first position of the preceding unprotected field.

New Line: Moves the cursor to the first position of the next or subsequent line that is in a nonbypass input field. If all the fields specified are bypass fields, the cursor moves to the position specified by the home address. If the home address is not specified, the cursor automatically moves to the first position of row 1.

Editing Keys

The editing keys are not operational if the keyboard is locked. These keys also cause an operator error if pressed after the Sys Req or Cmd key is pressed.

Del: Deletes the character above the cursor and moves the remaining characters in the field (must be an unprotected field) one position to the left. The last character position of the field is filled with a null character.

To delete a character, press and hold a shift key and then press the Del/Ins (Delete/Insert) key.

Ins: Sets the display station to insert mode. In this mode, characters can be inserted in a field (must be an unprotected field) while the character above the cursor and all following characters in the field are shifted 1 position to the right. Characters can be inserted as long as the rightmost character position in the field is a null character (hex 00).

Note: Insert mode must be established for each field; that is, the display station does not remain in insert mode for more than one field.

Erase Input: Clears all unprotected fields that have been modified by the operator, and moves the cursor to the position specified by the home address. To clear the unprotected fields, press and hold a shift key and then press the Erase Input/Home key.

Home: Moves the cursor to the position specified by the last home address. If there are no home addresses specified for the current screen, the cursor moves to the first position of row 1.

Error Reset: Lets the operator recover from a keyboard disabled condition. If there is no operator error condition at the time this key is pressed, this key will reset command mode, system request mode, or insert mode.

See *Error Reset (System Request Mode or Operator-Error Mode)* earlier in this chapter for more information on the Error Reset key.

Shift and Shift Lock: The two shift keys are each marked with a wide arrow pointing up, and the Shift Lock key is marked with a wide arrow pointing down. Pressing either shift key sets the keyboard in upper-shift for as long as the key is held down. Pressing a shift key also identifies the start of a shifted key function such as system request, delete, roll, and so on. The end of the shifted key function is identified by releasing the shift key.

When the Shift Lock key is pressed, the keyboard is set to upper-shift, and the keyboard remains in upper-shift after the Shift Lock key is released. To reset upper-shift, press (and release) one of the shift keys.

An inhibit downshift function permits the operation of some keys while a shift key is being held down without changing the shift state of the keyboard after the shift key is released. For example, if the keyboard is in upper-shift and a character is to be deleted, a shift key and the Del/Ins key must be pressed to delete the character. Then, when the shift key is released, the keyboard remains in upper-shift because of the inhibit downshift function.

The inhibit downshift function is enabled when any of the following occur (without an operator error):

- The keyboard is not locked, a shift key is pressed, and the Erase Input, Del/Ins, Roll Up, or Roll Down key is pressed.
- The keyboard is not locked, the Cmd key is pressed, a shift key is pressed and a valid command key is pressed.
- A shift key is pressed and then the Sys Req/Attn key is pressed.

The inhibit downshift function is reset if an operator error occurs, or if any key is pressed that does not enable the function.

Reverse Image: Reverses the image of the display screen if the keyboard is not locked. Normally, green letters appear on a black screen, but with the reverse image function the operator can display black letters on a green screen. To reverse the image, press the Cmd key, press and hold a shift key, and then press the \ key (see Figure 7-5).

Note: The \ key is the only command key that does not cause an AID byte to be sent to the host.

Special Field Keys

The special field keys include the Field Exit, Field-, Field+, and Dup keys. These keys:

- Are not operational if the keyboard is locked.
- Will cause operator errors if pressed after the Sys Req or Cmd key is pressed.
- Will not cause the cursor to leave a field if the display station is in insert mode.

Field Exit: Clears all of the characters from the current cursor position in a field to the end of the field, and moves the cursor to the first position of the next unprotected field. If the cursor is in any position of a mandatory fill field other than the first position, the cursor does not move and a four-digit error code is displayed. If the cursor is in the first position, the cursor moves to the first position of the next unprotected field. If the cursor is in a signed numeric field, this key causes a positive number to be generated. If the cursor is in a mandatory enter field that has not been modified, this key is not valid when pressed and causes a four-digit error code to be displayed.

Field-: Signals the end of an entered signed numeric field and causes a minus sign (-) to be displayed in the low-order character position of that field. This key is operational only in a signed numeric field.

Field+: Functions the same as the Field Exit key.

Dup: Causes a special character (an asterisk with a bar over it) to be displayed in the current cursor position and in all positions to the right of the cursor if the field is a dup-allowed field. This key also causes the cursor to move to the first position of the next unprotected field.

Note: The special character has a hex 1C value and is shown in Figure 7-2.

Display Station Modes

A display station can be in any one of several different modes. The following paragraphs describe each of these modes and how the modes are initialized.

Power On—No Mode Set

This mode is initialized when power to the display station is turned on but the display station is not operational. Characteristics of this mode are:

- The Ready indicator on the display station is on.
- Nothing is displayed on the screen except for the cursor which is positioned in the upper right-hand corner (row 1, column 80), and for the markers associated with the 5 indicators on the right side of the screen.
- Keystrokes are not serviced even though the Input Inhibited indicator is off.

Note: If the System Available indicator does not come on, System/34 is not powered on. If the System Available indicator comes on but the cursor stops in the upper right-hand corner of the display screen, the operator should turn the power off to the display station and then turn power back on. If the cursor again stops in the upper right-hand corner of the display screen, call your service representative.

Free Key Mode

Free key mode is initialized when power to the display station is turned on, and the following conditions are set:

- System/34 is powered on and operational.
- The serial interface between System/34 and the display station is operational.
- The system configuration has been specified.
- The display mode is set; that is, the display station is not in power on–no mode set and the cursor is moved to the upper left-hand corner (row 1, column 1) of the screen.

A display station remains in free key mode until the Sys Req/Enter keying sequence causes System/34 to write to the display station.

The first command issued to a display station in free key mode should be a Clear Unit command. This command is needed to reset field definitions that have been automatically set up for free key mode, and to clear the screen of any data that may have been entered in free key mode.

Characteristics of free key mode are:

- The display screen is specified as an alphameric enter field.
- Operator errors that are normally sensed by the work station controller are also sensed in free key mode, and the appropriate error code will be displayed.
- Keyboard functions handled by the work station controller operate normally except for those shown in Figure 7-7.
- Keyboard functions that generate an AID byte (except for Sys Req/Enter or Clear) cause error message 0099 to be displayed (see Figure 7-7). This error message is reserved for keyboard functions not allowed in free key mode.

System Request Mode

System request mode is initialized by pressing and holding a shift key, and then pressing the Sys Req/Attn key. This mode cannot be initialized if the display station is already in command key mode, operator-error mode, or insert mode.

System request mode is usually reset by pressing the Enter/Rec Adv key. However, if the Sys Req/Attn key is pressed by accident, this mode can be reset by pressing the Error Reset key.

Command Key Mode

Command key mode is initialized by pressing the Cmd key. This mode cannot be initialized if the display station is already in system request mode, insert mode, keyboard locked mode, or operator-error mode.

Command key mode is usually reset by pressing one of the valid command keys; that is, one of the top row keys as shown in Figure 7-5. However, if the Cmd key is pressed by accident, this mode can also be reset by pressing the Error Reset key.

	Function in Free Key Mode	
Keyboard Function (see Note 1)	**Operator Error Number**	**Free Key Error (0099)**
Print		X
Help		X
Roll Up		X
Roll Down		X
Enter/Rec Adv		X
Test Request		X
Attention		X
Commands 1-24		X
Field minus	0016	
Home (see Note 2)		X
Duplicate	0019	

Notes:
1. All other keyboard functions operate normally in free key mode.
2. The home function causes an error of 0099 only if the cursor is already home.

Figure 7-7. Key Functions in Free Key Mode

Keyboard Locked Mode

Keyboard locked mode is initialized for either of the following conditions:

- By pressing any AID request key.
- A system command is executed and the command forces the display station into this mode.

The keyboard locked mode cannot be reset from the keyboard. Instead, a system command is needed to reset this mode.

Insert Mode

Insert mode is initialized by pressing the Del/Ins key. This mode cannot be initialized if:

- The display station is in keyboard locked mode, command key mode, system request mode, or operator-error mode.
- The cursor is in a bypass or protected field.
- The Field Exit key needs to be pressed.

Insert mode is reset by pressing the Error Reset key, the Help key, or the Enter/Rec Adv key.

Operator-Error Mode

Operator-error mode is initialized by the work station controller when a rules violation is sensed during keystroke processing. For example, the operator presses an alphabetic key (instead of pressing a top row key) after pressing the Cmd key. A four-digit error code is displayed on the bottom row of the screen to inform the operator of the error. This mode can also be initialized by the system issuing a write error command.

This mode is reset by pressing the Error Reset key.

PROGRAMMING CHARACTERISTICS OF THE 5251 DISPLAY STATION

Program operation of each display station attached to System/34 is controlled by:

- An input/output block.
- A set of commands and orders that must be assembled by the users program.
- The appropriate supervisor call instruction.

Figure 7-8 shows the relationship of the host to the work stations and how the IOB, commands, and orders are passed between the two.

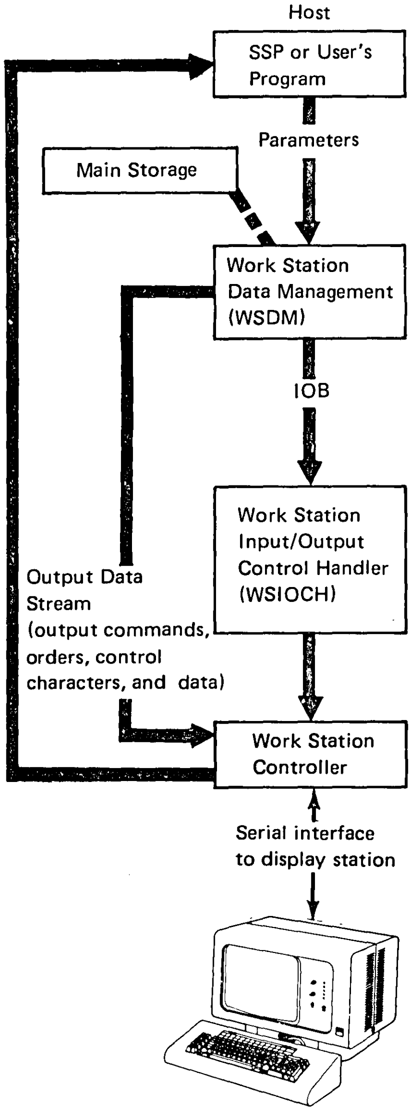

Host
SSP or User's Program
Parameters
Main Storage
Work Station Data Management (WSDM)
IOB
Work Station Input/Output Control Handler (WSIOCH)
Output Data Stream (output commands, orders, control characters, and data)
Work Station Controller
Serial interface to display station

Figure 7-8. Relationship of Host to the Work Stations

Display Station Input/Output Block

Each display station operation is specified by an input/output block (IOB) located in main storage. If more than one display station needs an operation to be performed, the IOBs are placed on a queue and a supervisor call instruction (see Chapter 3) is issued for each IOB. A supervisor call instruction cannot be issued for an IOB that is on a queue until the preceding IOB is posted complete; that is, until bit 1 of byte 1 in the IOB is set on.

The display station IOB is contained in the first 18 bytes of the terminal unit block. For purposes of this manual, and as shown in Figure 7-9, only the IOB is described. For a definition of each bit in the terminal unit block, see the *Data Areas Handbook*.

Displacement of Leftmost Byte in Hex	IBM Program Label	Length in Bytes in Decimal	Field Description (Note 1)
0	TUBECM	1	Event control mask The following bits must be set up before a supervisor call instruction is issued. *Hex* *Meaning* 80 No skip on general wait. 40 Data address is real. On = Task control block address points to real storage. Off = Task control block address points to the user's task control block for translated output. *Note:* The task control block address is in bytes 16 and 17 of the IOB. 20 Non I/O event.
1	TUBCOMPL	1	IOB completion code *Hex* *Meaning* 80 Active: Operation is in process and waiting completion. 40 Complete: Operation has been completed. 04 Input buffer assigned: The work station input/output control handler has successfully assigned the required input buffer. 01 Error detected: The requested operation could not be completed because of an error in the display station. Error status is posted in bytes 10 through 15 of this IOB.
2	TUBFLAG	1	Flag byte *Hex* *Meaning* 80 Report errors back to user: The system error recovery procedures are suppressed and the errors are reported to the user. 20 Automatic input buffer assignment. Successful assignment of an input buffer is indicated in the completion code (bit 5). 10 Terminal unit block not allowed off vertical TUB chain: Indicates that the operation is complete, and the TUB can be pulled off the verticle TUB chain. 08 Device is online: This bit must be on before the system can accept any interrupts from the display stations.

Figure 7-9 (Part 1 of 3). Display Station IOB

Displacement of Leftmost Byte in Hex	IBM Program Label	Length in Bytes in Decimal	Field Description (Note 1)
2 (continued)	TUBFLAG		*Hex Meaning* 04 Read input command is issued to the terminal unit block. 02 Read input command is not completed. When the work station controller finishes the read input operation, this bit is turned off by the work station input/output control handler, and if the terminal unit block is posted complete, bit 5 will also be turned off by the work station input/output control handler. 01 Data is in control storage.
3	TUBCMND	1	Command code Bits 0-4 Work station controller address (11000) Bit 5 Must be 0 Bits 6-7 Set to indicate the operation 00 = Execute 01 = Invite (allow interrupts from work stations) 10 = Quiesce printer 11 = Cancel (used to transfer from invite to execute)
4	TUBCMOD	1	Command modifier: Specifies the operation to be executed. *Hex Command* A7 Output/invite. This is a combined function of output (hex 27) and invite (as defined in the command code byte). 62 Read screen. 42 Read input fields. 27 Output (causes the output data stream to be read by System/34). 06 Save screen. 02 Save tables.
5	TUBINIT@	1	Device address Bits 0-3 Twinaxial cable address. Bit 4 Must be 0. Bits 5-7 Work station address. (The address of each work station is set with toggle switches on each work station.)
6	TUBDATA@	2	Data buffer address: The logical address of the data buffer to be used for execute commands. The data buffer can be in main storage or in control storage as indicated by bit 7 of byte 2 of this IOB.

Figure 7-9 (Part 2 of 3). Display Station IOB

Displacement of Leftmost Byte in Hex	IBM Program Label	Length in Bytes in Decimal	Field Description (Note 1)
8	TUBCOUNT	2	Data buffer length (byte count): The exact length, in bytes, to be sent to the work station. Length values greater than 4096 cause error completion status.
A	TUBSENS0	1	Status byte 0 (Note 2)
B	TUBSENS1	1	Status byte 1 (Note 2)
C	TUBSENS2	1	Status byte 2 (Note 2)
D	TUBSENS3	1	Status byte 3 or attention identification (AID) byte (Note 3)
E	TUBSENS4	1	Status byte 4 (Note 2)
F	TUBSENS5	1	Status byte 5 (Note 2)
10	TUBTCB@	2	Task control block address for data buffer address translation registers.

Notes:

1. Unused and reserved bits, or bits not used by the work station input/output control handler are not shown in the *Field Description* column.
2. See *Display Station Status Bytes and Error Recovery* in this chapter for a description of the status bytes.
3. See Figure 7-3 for the hexadecimal representations of the AID byte.

Figure 7-9 (Part 3 of 3). Display Station IOB

Read Commands and Control Commands

System/34 controls the display stations with a set of commands that are sent to the work station controller (see Figure 7-8). The read commands are sent in the command modifier byte (byte 4 of the IOB), the control commands are defined by bits in the terminal unit block, and the output commands are sent in the output data stream.

The four read commands and two control commands are described in the following paragraphs; the output commands are described under *Output Commands* later in this chapter. The hexadecimal value of each read command is given in parentheses after the name of the command.

Save Tables (Hex 02)

The save tables command lets System/34 save the control information and field definitions for any display station. Then, if needed at a later time, the data can be restored by the restore tables command.

Characteristics of the AIDs when this command is executed are:

- Any pending AID is also saved (except for Sys Req and Attn) but is not reset when the command is completed.
- The AID is established again when the data is restored.
- The AID byte must not be processed twice.

CAUTION
The data received with this command must not be modified in any way. Failure to follow this rule can result in unpredictable work station controller operations when the data is restored.

The format of the data restored is:

Bytes	**Description**
0 and 1	Hex 0401, which is the format of the restore tables command; it permits the data to be restored with no modifications or additions by the user.
2 and 3	Length of the data (in bytes) sent in response to the save tables command, including bytes 0 through 3. When executing the save tables command, the byte count in IOB bytes 8 and 9 must be equal to, or greater than, the number of bytes sent. Use this value when generating the byte count for the restore tables command.
4-67	Display station parameters (generated by the work station controller).
68-579	Format table of 512 bytes for the display station

This command is not valid if the display station is offline, is not powered on, or is in system request or hardware-error mode.

The save tables command causes the keyboard to lock before the command is executed, and all keystrokes are ignored (including those normally processed when the keyboard is locked) until a save screen or clear unit command is executed. The save screen or clear unit commands are the only commands that should be sent to a display station after the save tables command is executed.

Save Screen (Hex 06)

The save screen command lets System/34 save the data on the current screen. Then, if needed at a later time, the data can be restored by the restore screen command. This command must follow a save tables command directed to the same display station. The command sequence of save tables followed by save screen is needed to save all the status (control information and data) of a display station. This command sequence does not affect the state of the display station; that is, the state that the display station was in before the save tables command was executed.

CAUTION
The data received with this command must not be modified in any way. Failure to follow this rule can result in unpredictable work station controller operations when the data is restored.

The format of the data restored is:

Bytes **Description**

0 and 1 Hex 0405, which is the format of the restore screen command; it permits the data to be restored with no modifications or additions by the user.

2 and 3 Length of the data (in bytes) sent in response to the save screen command, but excluding bytes 0 through 3. When executing the save screen command, the byte count in IOB bytes 8 and 9 must be equal to, or greater than, the number of bytes sent. Change this value to include bytes 0 through 3 when generating the byte count for the restore screen command.

4-1043 Displayed data for 12 rows.

4-2003 Displayed data for 24 rows.

This command is not valid if the display station is offline, is not powered on, or is in hardware-error mode; or if the Save Table command was not issued before the save screen command was issued.

Note: If an error is sensed while the save screen command is being executed, the error retry process must start with the save tables command.

Read Input Fields (Hex 42)

The read input fields command causes the work station controller to move the data from all fields specified in the format table back to System/34. The fields are sent in the sequence in which they are specified on the display screen unless the user has selected a different sequence. A different sequence is selected by using the start of header order and resequencing field control words. Both are described in this chapter.

Any nulls characters sensed in the fields are converted to blank characters. Therefore, the field is filled to its original length and the field is sent immediately following the preceding field without any intervening control information.

If the field is specified as a signed numeric field, the last character in the field is never sent to the system. In addition, if the last character of a signed numeric field is a minus sign, the zoned portion of the second to last character in the field is changed to a hex D. If the last character is not a minus sign, the second to last character is not changed.

The read input fields command is not valid if the display station is offline, is not powered on, or is in hardware-error or operator-error mode. Also, the command is not valid if the keyboard is not locked, or if the format table is not loaded.

Note: Any pending AID byte, except for hex F2 (Attn) or hex F0 (Sys Req/Enter), is reset when this command is completed.

Read Screen (Hex 62)

The read screen command causes all data on the display screen to be sent to System/34. The data is sent in the sequence in which it appears on the display screen; that is, the first row is sent first, the second row is sent next, and so on. The data includes all fields (input and noninput) and all display attributes.

The byte count in IOB bytes 8 and 9 must be equal to, or greater than, the number of bytes sent. An error is posted for byte counts that are too small.

The read screen command is not valid if the display station is offline, is not powered on, or is in hardware-error or operator-error mode.

Note: Any pending AID byte, except for Attn or Sys Req/Enter, is reset when this command is completed.

Set Operator Alert Indicators

The set-operator-alert-indicators command is one of the two control commands; it lets the system inform the operator that attention is needed by:

- Sounding the audible alarm on the display station for approximately 1 second.
- Turning on the Message Waiting indicator on the display station.

This command is not valid if the display station is offline, is not powered on, or is in hardware-error mode.

Note: Any pending AID byte is not reset by this command.

Reset Operator Alert Indicators

The reset-operator-alert-indicators command is one of the two control commands; it lets the system turn off the Message Waiting indicator on the display station. The audible alarm, set by the set-operator-alert-indicators command, is not affected by this command. Instead, the audible alarm is automatically reset.

This command is not valid if the display station is offline, is not powered on, or is in hardware-error mode.

Note: Any pending AID byte is not reset by this command.

Display Station Output Data Stream

The output data stream (see Figure 7-10) sent to the work station controller includes data and control information. The control information includes two write control characters, five orders, and seven output commands as described in the following paragraphs. The two write control characters are described under *Write Data (Hex 11).*

Orders

The following paragraphs describe the five orders and the associated data that are a part of the write data command in the output data stream. The hexadecimal value of each order is given in parentheses after the name of the order.

Figure 7-10 shows the combination of orders that follow the write data command byte in the output data stream.

Start of Header (Hex 01): Selects the resequencing function when data is read from the display screen. Four data bytes follow this order in the output data stream. These four bytes are:

Byte	Meaning
1	Set to hex 03.
2	Not used.
3	Not used.
4	Set to hex 00 if the displayed data is to be returned to the system in the same sequence as it appears on the screen. If set to a value other than hex 00 the displayed data will be returned to the system in a sequence determined by the resequencing field control words in the format table. The value of this byte also points to the first field to be returned by the read input fields command.

Either of the following conditions associated with the start of header order will cause an error:

- The output data stream ends before the number of bytes needed have been sent.
- The first byte after the order is not hex 03.

Repeat to Address (Hex 02): Causes a character to be displayed in every position on the display screen starting in the position specified by the current buffer address to (and including) the last position specified in this order. Only one character is displayed if the position specified in this order is equal to the position specified by the current buffer address. The current buffer address is then updated to the value of the last position plus 1.

The last position and the character displayed are specified in the three bytes that follow this order in the output data stream. These three bytes are:

Byte	Meaning
1	Row address.
2	Column address.
3	Character that is displayed.

Any of the following conditions associated with the repeat to address order will cause an error:

- The output data stream ends before the number of bytes needed have been sent
- A row address value of 0 or a value greater than 24.
- A column address value of 0 or a value greater than 80.

Set Buffer Address (Hex 11): Determines where the displayed data will start on the display screen as instructed by the two address bytes that follow this order in the output data stream. The first address byte is the row address and the second address byte is the column address.

When this order is not in the output data stream, the displayed data starts in row 1, column 1 because the write data command initializes the buffer address to the start of the display screen.

Any of the following conditions associated with the set buffer address order will cause an error:

- The output data stream ends before the number of bytes needed have been sent.
- A row address value of 0 or a value greater than 24.
- A column address value of 0 or a value greater than 80.

Insert Cursor (Hex 13): Establishes the home address and inserts the cursor in a position on the display screen as instructed by the two address bytes that follow this order in the output data stream. The first address byte is the row address and the second address byte is the column address.

If the keyboard is locked when the write data command is issued, and if the write data command causes the keyboard to unlock, the cursor is moved to the location specified by this order. If the keyboard does not unlock, the cursor does not move.

If more than one insert cursor order is in the output data stream, only the last one is used. The position specified is saved and it becomes the home address for the Erase Input/Home function.

Any of the following conditions associated with the insert cursor order will cause an error:

- The output data stream ends before the number of bytes needed have been sent.
- A row address value of 0 or a value greater than 24.
- A column address value of 0 or a value greater than 80.

Start of Field (Hex 1D): Resets any pending AID bytes (except Attn or Sys Req/Enter), specifies an input field, and causes the characteristics of the input field to be saved in the format table (see *Format Table* later in this chapter) as instructed by the information that follows this order in the output data stream. The sequence of this information is:

1. A 2-byte field format word (see *Field Format Word* later in this chapter).

2. Up to two 2-byte field control words which are optional. (See *Field Control Word* later in this chapter.)

3. A 1-byte display attribute pertaining to the input field. The bits in the display attribute byte and the meaning for each bit (when on) are:

Bit	Meaning
0-2	Set to 001
3	Column mark on
4	Blink field
5	Underscore field
6	Select high intensity
7	Select reverse image (black on green) if this bit is on, or select normal image (green on black) if this bit is off.

Note: If bits 5, 6, and 7 are all on, the input field is not displayed.

4. Two length bytes that determine how many character positions are reserved for this field on the display screen.

Any of the following conditions associated with the start of field order will cause an error:

- The output data stream ends before the needed number of bytes have been sent.

- More than 8 bytes separate the order and the display attribute.

- The field length is zero.

- The address for the end of the field is past the end of the display screen.

- Input field addresses are not in an increasing order. This pertains to input fields that were specified by a preceding write data command. No input field can be specified if its starting address is less than, or equal to, the starting address of an input field already specified.

- Too many input fields have been specified for the display screen.

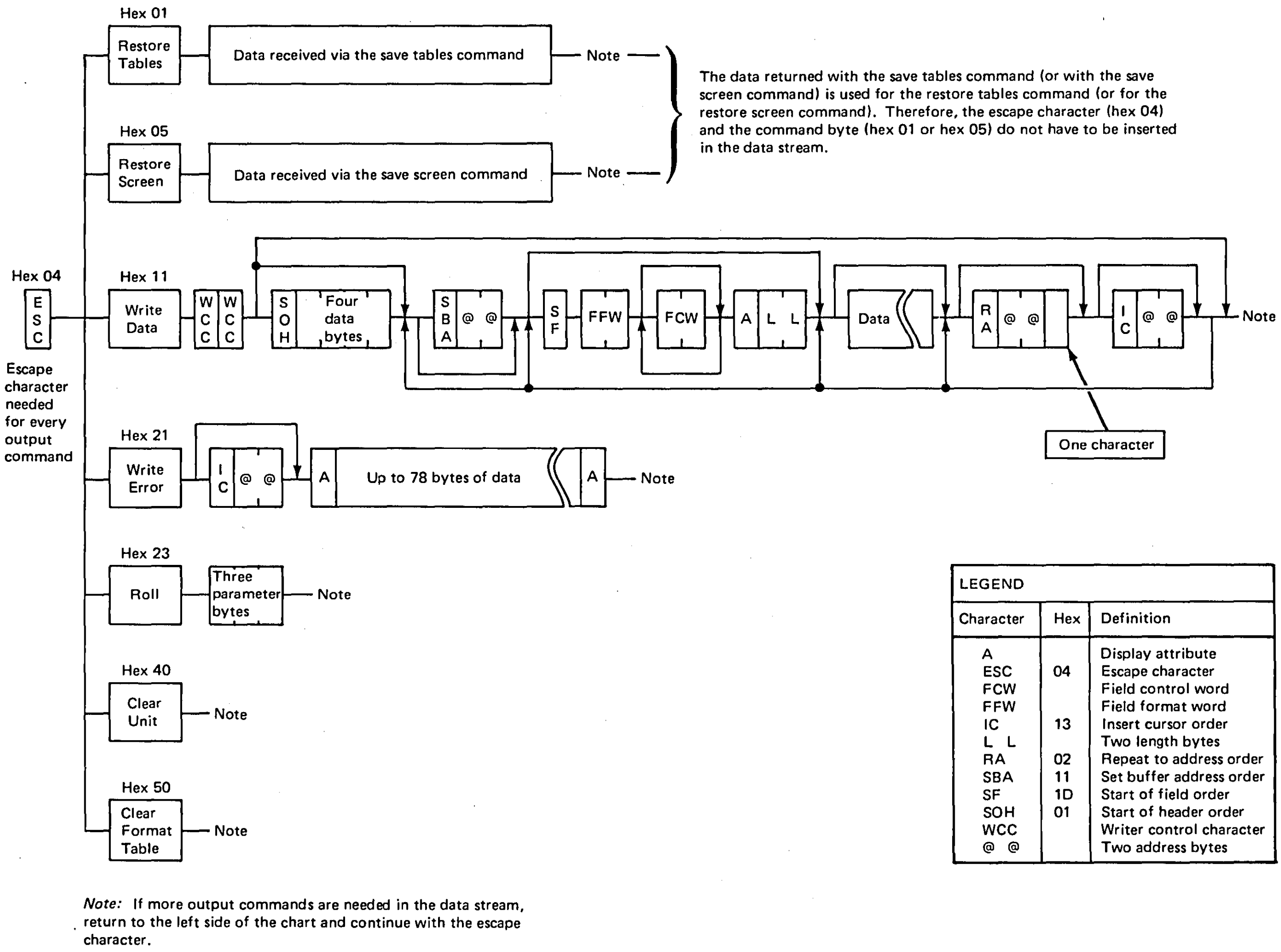
The following information should be considered when using the start of field order:

- The order reserves character positions on the display screen and these positions are not modified.

- The display attribute needs one byte on the display screen but this position is not included in the length bytes. The display attribute is placed at the position specified in the current buffer address.

- The first character position of an input field is at the position specified by the address counter plus 1. After the order is executed, the current buffer address points to this position.

- An ending display attribute of hex 20 is written to the display screen after the input field being specified. Therefore, two display attributes are written for each input field and they are separated by the number of character positions indicated in the length bytes.


Hex 04
E S C
Escape character needed for every output command
Hex 01
Restore Tables
Data received via the save tables command
Note
Hex 05
Restore Screen
Data received via the save screen command
Note
The data returned with the save tables command (or with the save screen command) is used for the restore tables command (or for the restore screen command). Therefore, the escape character (hex 04) and the command byte (hex 01 or hex 05) do not have to be inserted in the data stream.
Hex 11
Write Data
W C C
W C C
S O H
Four data bytes
S B A
@ @
S F
FFW
FCW
A
L L
Data
R A
@ @
I C
@ @
Note
One character
Hex 21
Write Error
I C
@ @
A
Up to 78 bytes of data
A
Note
Hex 23
Roll
Three parameter bytes
Note
Hex 40
Clear Unit
Note
Hex 50
Clear Format Table
Note

LEGEND		
Character	Hex	Definition
A		Display attribute
ESC	04	Escape character
FCW		Field control word
FFW		Field format word
IC	13	Insert cursor order
L L		Two length bytes
RA	02	Repeat to address order
SBA	11	Set buffer address order
SF	1D	Start of field order
SOH	01	Start of header order
WCC		Writer control character
@ @		Two address bytes

Note: If more output commands are needed in the data stream, return to the left side of the chart and continue with the escape character.

Figure 7-10. Display Station Output Data Stream

Output Commands

When byte 4 of the IOB (command modifier) is equal to hex 27 or hex A7, the output data stream (containing the output commands) is moved from main storage to the work station controller. Then the work station controller executes the output commands in the order in which they appear in the data stream. The data stream can have one or more output commands but cannot have any of the read commands or control commands (see *Read Commands and Control Commands* earlier in this chapter.)

In the following descriptions of the output commands, the hexadecimal value of each output command is given in parentheses after the name of the command.

Restore Tables (Hex 01): Restores the contents of the format table and the state of the display station by returning the same data (received via the save tables command) to the work station controller. The data restored must be directed to the display station that the data originated from. After this command is executed, the clear unit command or the restore screen command are the only commands that should be sent to the display station.

This command is not valid if the display station is offline, is not powered on, or is in hardware-error or system request mode.

Restore Screen (Hex 05): Restores the contents of the display screen by returning the same data (received via the save screen command) to the work station controller. The data restored must be directed to the display station that the data originated from.

The restore screen command must follow the restore tables command in the output data stream, and both commands must be directed to the same display station. The command sequence of restore tables followed by restore screen is needed to save all the status of a display station. If an error is sensed while executing the restore screen command, the error retry process must start with the restore tables command.

This command is not valid if the display station is offline, is not powered on, is in hardware-error mode, or if a restore tables command was not directed to the same display station before issuing this command.

Write Data (Hex 11): Writes data to the display screen, unlocks or locks the keyboard, sets various display screen modes, and assembles or adds to a format table (see *Format Table* later in this chapter). To do all these functions the command byte (hex 11) is followed by two write control characters that are followed by orders and other data as shown in Figure 7-10. See *Orders* earlier in this chapter for a description of the orders.

Data bytes are defined as any character following the write data command in the output data stream if the character is not a write control character, escape character, order, or a parameter associated with an order. Each character is checked to determine if it is an order or an escape character and if the character is neither, the character is written to the display screen. The data written to the display screen includes all data bytes and all display attributes. The data is written to the display screen at the present value of the address counter, and the address counter is increased by a value of 1 for every byte written.

The write control characters must immediately follow each write data command in the output data stream to select specific operations for the display stations to perform. The operations selected in the first write control character are performed as soon as the work station controller receives this character. The operations selected in the second write control character are not performed until all information associated with the write data command is processed.

The bits in the first write control character and the meaning for each bit (when on) are:

Bit	**Meaning**
0	Write null characters to all input fields (except bypass fields) that have the modified-data-tag bit set on (bit 4 of the field format word), lock the keyboard, and reset any pending AID byte (except Attn or Sys Req/Enter).
1	Reset the modified-data-tag bit in all input fields (except bypass fields), lock the keyboard, and reset any pending AID byte (except Attn or Sys Req/Enter).
2-7	Reserved.

The bits in the second write control character and the meaning for each bit (when on) are:

Bit	**Meaning**
0	Reserved.
1	Reserved.
2	Reset blinking cursor.
3	Set blinking cursor. *Note:* If bits 2 and 3 are both on, set the blinking cursor.
4	Unlock keyboard and reset any pending AID byte (except Attn or Sys Req/Enter).
5	Sound alarm.
6	Set Message Waiting indicator off.
7	Set Message Waiting indicator on. *Note:* If bits 6 and 7 are both on, set the Message Waiting indicator on.

The end of the data stream for a write data command is determined in one of two ways:

- By sensing another escape character (hex 04) which would indicate the start of a new command.
- By a byte count in the IOB (bytes 8 and 9) reaching a value of 0.

The write data command is not valid if the display station is offline, is not powered on, is in hardware-error mode, or is in operator-error mode.

Write Error (Hex 21): Forces the display station into operator-error mode, or it lets the operator reset operator-error mode. For example, if the display station is in operator-error mode and the Help key has been pressed, the write error command must be issued by the system to let the operator reset operator-error mode.

When a display station receives this command, the following occurs:

- Display station is forced into operator-error mode (command key mode, insert mode, and system request mode are reset).
- Keyboard is locked.
- Cursor is made to blink.
- The last line of the display screen is saved, but is not changed.
- The cursor is moved if an insert cursor order follows the write error command in the output data stream.
- The data following the write error command in the output data stream is displayed on the last row of the display screen.

If the command byte (hex 21) is followed in the output data stream by an insert cursor order and two address bytes (all three are optional), the format of this command is:

1. Escape character (hex 04).
2. Command byte (hex 21).
3. Insert cursor order (hex 13); this order and the two address bytes described in number 4 are optional. If the insert cursor order immediately follows the command byte, the first four data bytes in number 6 generate the error code. The error code is sent to the host if the Help key is pressed. If the insert cursor order does not immediately follow the command byte, the fourth, fifth, sixth, and seventh data bytes following the last insert cursor order generate the error code.
4. Two address bytes (row address followed by column address) that determine where the cursor is moved to.
5. Display attribute selected by the user. To be consistent with operator-error mode, this attribute should be hex 2A to select high intensity and blink field. However, if the Help key is pressed when the display station is in operator-error mode, the work station controller forces an attribute of hex 22 (high intensity and no blinking) into the first position of the last line.
6. Up to 78 data bytes. The total number of data bytes (including the 2 display attributes) cannot be more than 80.
7. Display attribute selected by the user. To be consistent with operator-error mode, this attribute should be hex 27 to select nondisplay.

Note: No checks are made to verify the presence of the two display attributes. For a description of the display attributes, see *Start of Field (Hex 1D)* earlier in this chapter.

The write error command is not valid if the display station is offline, is not powered on, is in hardware-error mode, or if the Help key is not pressed after the display station is in operator-error mode.

Roll (Hex 23): Causes the message on the display screen to roll up or roll down as determined by the 3 parameter bytes that follow the command byte (hex 23) in the data stream. The 3 parameter bytes are:

Byte	Bits	Meaning
1	0	If on, roll down. If off, roll up. All 80 character positions of each line are rolled, including the display attribute bytes.
1	1 and 2	00
1	3-7	Number of lines rolled (0 = no roll).
2	0-7	Number specifying the top line of the information rolled. This number cannot be 0.
3	0-7	Number specifying the bottom line of the information rolled. This number must be greater than the number specified in byte 2 for the top line

The roll command is not valid if the display station is offline, is not powered on, is in hardware-error mode, or is in operator-error mode.

Any of the following conditions will cause an error:

- There are less than 3 bytes in the data stream after the command byte (hex 23).
- The number specifying the top line of the roll is 0.
- The number specifying the bottom line is greater than 24, or it is less than, or equal to, the number specifying the top line.
- The roll magnitude is greater than the value obtained by subtracting the top line number from the bottom line number.

The format table is not affected by this command. It is the user's responsibility to be sure that the display corresponds to the format table after the roll is completed.

Note: Lines rolled out of the roll area are lost, and lines vacated by the roll are not modified.

Clear Unit (Hex 40): Performs the following functions for the addressed display station:

- Clears the information on the display screen and the format table.
- Clears any pending AID byte.
- Locks the keyboard.
- Moves the cursor to row 1, column 2.
- Sets the home address to row 1, column 1.
- Sets the initial resequence field number to 0.
- Resets the system request, command key, operator-error, and insert mode.
- Resets the master modified data tag bit.
- Writes a display attribute of hex 20 to row 1, column 1.

The clear unit command should be executed as the first command of any new job to ensure that any preceding data is cleared from the display screen and also from the format tables. This command is not valid if the display station is offline, is not powered on, or is in hardware-error mode.

Clear Format Table (Hex 50): Performs the same functions as the clear unit command (as described in the preceding paragraph) but does not (1) clear the information on the display screen, (2) clear operator-error mode, or (3) clear system request mode. This command is not valid if the display station is offline, is not powered on, or is in hardware-error or operator-error mode.

Format Table

The format table contains an entry for each input field specified on the display screen. All fields specified in the format table are returned to the system when a read input fields command is issued. Input fields can be coded so that they cannot be modified from the keyboard.

Each entry in the format table must include a 2-byte field address word and a 2-byte field format word. Another word, a 2-byte field control word, is optional; the entry can have up to two field control words. Therefore, each entry in the format table is at least 4 bytes long but can be as many as 6 or 8 bytes long. The format table is less than, or equal to, 512 bytes.

To determine the number of input fields permitted on the display screen, use this calculation: 4 (total number of input fields) plus 2 (number of input fields needing resequencing field control words) plus 2 (number of input fields needing modulus 10/11 check digit verification). The total number must be less than 512 bytes.

Field Address Word

A 2-byte field address word is generated by the work station controller for each input field. This word determines the starting address of the input field on the display screen.

Field Format Word

A 2-byte field format word is generated by the user's program and is sent to the work station controller via the output data stream. This word determines the characteristics of the input field and is shown in Figure 7-11.

See *Input Fields* later in this chapter for a description of the different types of input fields.

Field Control Word

A 2-byte field control word is not always needed in the format table entry when describing the input fields, but any format table entry can have as many as two of these words. If used, the field control words are generated by the user's program and sent to the work station controller via the output data stream.

Two types of field control words are used; the resequencing type and the check-digit type. When both types are used, the resequencing type must be the first one in the data stream.

Resequencing Field Control Word: Identifies the sequence in which the input fields are sent to the system when a read input fields command is issued. When the input fields are not resequenced, they are sent to the system in the sequence in which they appear on the display screen.

If the input fields are resequenced, the first field sent to the system is identified with the start of header order, and the last field must have bits 8 through 15 of the field control word set to hex FF. The bits of the resequencing field control word are as follows:

Bits **Meaning**

0 and 1 Set to 10.

2 and 3 Set to 00.

4-7 Set to 0000.

8-15 Indicates the next field returned to the system. For example, assume that 10 input fields are specified on the display screen and they are sequentially numbered 1 through 10. However, they are to be resequenced in the order 5, 6, 2, 3, 4, 9, 10, 1, 7, and 8 when the read input fields command is issued. In this case:

- The fourth byte of the start of header order would contain hex 05 (indicating the first field returned to the system).

- Input fields 2, 3, 5, 7, and 9 do not need resequencing field control words because the next sequential fields are returned next; that is, 3 follows 2, 6 follows 5, etc.

- Input fields 1, 4, 6, 8, and 10 need a resequencing field word as follows:

Field	**Field control word in hex**
1	8007 (field 7 follows field 1)
4	8009 (field 9 follows field 4)
6	8002 (field 2 follows field 6)
8	80FF (field 8 is the last input field in this example)
10	8001 (field 1 follows field 10)

Bits	Input Field Attribute and Description
0-1	Set to 01 to indicate a field format word.
2	Bypass bit 0 = Not a bypass field. 1 = A bypass field: The operator cannot key data into this area of the display screen, and the cursor is positioned in the first position of the next input field that isn't a bypass field.
3	Dup key enabled 0 = Dup key cannot be used in this field. 1 = Dup key can be used in this field.
4	Modified data tag (MDT) bit 0 = This field has not been modified. 1 = This field has been modified.
5-7	Field description 000 = Alphameric: All characters allowed. 001 = Alphabetic only: Allows uppercase and lowercase letters, comma, period, hyphen, blank, and Dup keys. 011 = Numeric only: Allows 0-9, comma, period, +, -, blank, and Dup keys. 100 = Katakana (Japan only). 111 = Signed numeric: Allows 0-9 and Dup keys. *Note:* All remaining bit configurations are reserved.
8	Automatic enter 0 = No automatic enter. 1 = Automatic enter: System assumes that the Enter/Rec Adv key was pressed after the cursor left the input field via the Field Exit key, Field+ key, Field- key, or by entering the rightmost character into the field.
9	Field Exit key needed 0 = Cursor leaves field when rightmost character is entered. 1 = Cursor does not leave field when rightmost character is entered.
10	Monocase mode 0 = Display lowercase a through z. 1 = Convert lowercase a through z to uppercase.

Figure 7-11 (Part 1 of 2). Field Format Word

Bits	Input Field Attribute and Description
11	Reserved
12	Mandatory enter 0 = Not a mandatory enter field. 1 = Is a mandatory enter field; at least 1 character must be entered.
13-15	Adjust or fill option 000 = No adjust specified. 101 = Right adjust and 0 fill: The rightmost character entered in this field is shifted to the right boundary of the field. The positions to the left of the characters entered are filled with zeros. 110 = Right adjust and blank fill: The rightmost character entered in this field is shifted to the right boundary of the field. The positions to the left of the characters entered are filled with blanks. 111 = Mandatory fill: Requires that if one position of the field is filled, then all positions must be filled. The Dup key satisfies this requirement and causes an exit to the next input field. *Note:* All remaining bit configurations are reserved.

Figure 7-11 (Part 2 of 2). Field Format Word

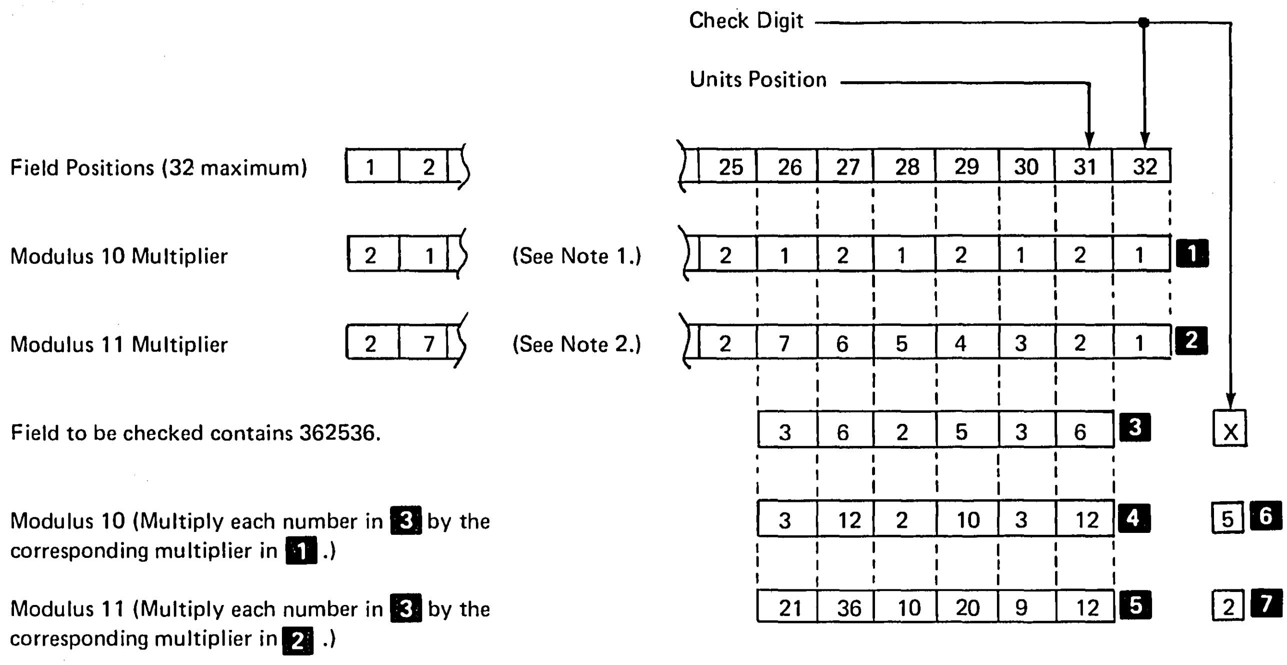
Check-Digit Field Control Word: Specifies that an input field is to be checked for its validity using modulus 10 or modulus 11 checking. This word is assembled as follows:

Bits	Meaning
0 and 1	10.
2 and 3	11.
4-7	0001.
8-10	Self check operation where bits 8 through 10 are set to 010 for modulus 11 checking, or set to 101 for modulus 10 checking.
11-15	00000.

The following rules apply when input fields are checked:

- Signed-numeric, numeric only, alphabetic only, and alphanumeric fields can be specified for checking.
- Field lengths are limited to 32 character positions. Signed-numeric fields can be 33 positions, but the signed position is not checked.
- The check digit is the rightmost character in the field.
- Only the four low-order bits of each character are used for the check digit.
- The number to use in calculating the check digit for non-numeric characters is determined by the EBCDIC representation for the character.
  - A is equal to C1; the number to use is 1
  - R is equal to D9; the number to use is 9.
  - % is equal to 6C; the number to use is 0. The number to use is 0 when the four low-order bits are either A, B, C, D, E, or F.
- If an input field does not check correctly, an error code of 0015 is displayed, and the cursor is moved to the first position of the field in error.

Figure 7-12 shows the calculations of the check digit used for modulus 10 and modulus 11 checking.



Field Positions (32 maximum)	1	2		25	26	27	28	29	30	31	32			
Modulus 10 Multiplier	2	1	(See Note 1.)	2	1	2	1	2	1	2	1	1		
Modulus 11 Multiplier	2	7	(See Note 2.)	2	7	6	5	4	3	2	1	2		
Field to be checked contains 362536.					3	6	2	5	3	6		3	X	
Modulus 10 (Multiply each number in 3 by the corresponding multiplier in 1.)					3	12	2	10	3	12		4	5	6
Modulus 11 (Multiply each number in 3 by the corresponding multiplier in 2.)					21	36	10	20	9	12		5	2	7

For modulus 10:

1. Add the *digits* in 4 (3 + 1 + 2 + 2 + 1 + 0 + 3 + 1 + 2 = 15). If the sum ends in 0, the check digit is 0.
2. Subtract the sum from the next higher number ending in 0. (20 - 15 = 5)
3. In this example, the check digit is 5. 6

For modulus 11:

1. Add the *numbers* in 5 (21 + 36 + 10 + 20 + 9 + 12 = 108).
2. Divide the sum by 11 and record the remainder. If the remainder is 1, the number being checked cannot be used. 108/11 = 9 (with a remainder of 9)
3. Subtract the remainder from 11 (11 - 9 = 2).
4. In this example, the check digit is 2. 7

Notes:
1. The modulus 10 multipliers are 2 and 1 for every other position in the field.
2. The modulus 11 multipliers are 7, 6, 5, 4, 3, and 2 for positions 2 through 7, 8 through 13, 14 through 19, 20 through 25, and 26 through 31. The multiplier for position 1 is 2.

Figure 7-12. Calculation of Modulus 10/Modulus 11 Check Digit

Input Fields

The user must specify the input fields that are modified from the keyboard and then sent from the display stations to an application or utility program. Bits 5 through 7 of the field format word specify the type of input field, and the remaining bits of the field format word specify the characteristics of the input field.

Types of Input Fields

There are five types of input fields as described in the following paragraphs. The decode of bits 5 through 7 of the field format word determine the type of input field; the decode of these bits is given in parentheses after the name of the input field.

Alphameric (000): Lets the operator enter any lowercase or uppercase character (alphabetic, numeric, or special) from the keyboard. The shift keys override the lowercase letters (and numbers) and force uppercase letters (and special characters).

Alphabetic Only (001): Lets the operator enter any lowercase or uppercase letter and blank, comma, period, hyphen, or dup (if the dup allowed bit is on). An error occurs if the operator tries to enter any other character.

Numeric Only (011): Lets the operator enter any number (0-9) and blank, plus, minus, comma, period, or dup (if the dup allowed bit is on). An error occurs if the operator tries to enter any other character.

Katakana (100): This type of input field is used only in Japan. If Katakana is specified for a display station that does not support Katakana, the input field type defaults to alphameric.

Signed Numeric (111): Lets the operator enter any number (0-9) and dup (if the dup allowed bit is on). An error occurs if the operator tries to enter any other character.

Signed numeric fields are always right-adjusted and the rightmost field position is always reserved for the sign (minus for negative and blank for positive). Therefore, the maximum number of characters that can be entered is 1 less than the field length.

If the Field- key is used to exit the field, a minus sign is placed in the rightmost field position. If the Field+ or Field Exit key is used to exit the field, a blank is placed in the rightmost field position. The rightmost field position is not returned when a read input field command is issued.

Characteristics of Input Fields

Bits 2 through 4 and 8 through 15 of the field format word specify the characteristics of the input fields. Bit 4 is the modified data tag bit, bit 10 is the uppercase bit, and bit 11 is reserved. The remaining bits are described in the following paragraphs and the bit numbers are given in parentheses after the name of the input field characteristic.

Bypass (Bit 2): Prevents the operator from entering data into a specified input field. Bypass also causes the cursor to move to the first position of the next nonbypass input field.

Note: Bypass specified input fields are returned to the system when a Read Input Fields command is issued.

Dup Allowed (Bit 3): Allows the Dup key function in this field.

Automatic Enter (Bit 8): Causes an enter AID to be sent to the host when the cursor is in the input field and:

- the Field Exit or Field+ key is pressed.
- the Field- key is pressed in a signed numeric field.
- the Dup key is pressed and the dup allowed bit is on.
- a character is entered into the low-order character position of the input field and field exit control is not enabled.

Field Exit Control (Bit 9): The operator must press a nondata type key before the cursor will leave the input field.

Mandatory Enter (Bit 12): The modified data tag bit associated with the input field must be set on before the information on the display screen can be returned to the system This is normally accomplished by entering at least one data character into the field from the keyboard.

When any automatic enter function is activated, all fields on the display screen are checked to see if they are specified as mandatory enter fields. If any specified mandatory enter field does not have its modified data tag bit set on, the automatic enter function is not executed. Instead, the cursor is positioned in the first position of the first mandatory enter field that does not have its modified data tag bit set on, and a four-digit error code of 0007 is displayed.

When the cursor is in the first position of a specified mandatory enter field, the Field Exit, Field+, and Field- keys will not cause the cursor to move until at least one character has been entered. A four-digit error code of 0021 is displayed if any of these keys are pressed when the cursor is in the first position of the field.

Mandatory Fill (Bits 13-15): If one position of the input field is filled, then all positions must be filled. The Dup key satisfies this condition and causes the cursor to move to the first position of the next input field. Bits 13-15 of the field format word are set to 111 for a mandatory fill field.

Other rules that apply when the cursor is in a mandatory fill field are:

- If no characters have been entered in this field, the cursor can be moved from this field by pressing the Field Advance, Field Backspace, Field Exit, Field+, Field-, Erase Input/Home, any of the cursor motion keys, or any AID generating key that locks the keyboard.

  Note: The Field Exit, Field+, and Field- keys will cause the cursor to move only from the first position of a mandatory fill input field. If the Field- key is pressed when the cursor is in the first position of a mandatory fill input field (but the field is not a signed numeric field), a four-digit error code of 0016 is displayed.

- If some characters have been entered in this field and the field contains null characters, the cursor motion keys can be used unless they would cause the cursor to leave this field. A four-digit user code of 0014 is displayed if the cursor motion keys cause the cursor to leave this field.

Right-Adjust (Bits 13-15): Causes the data entered in this field to shift to the right until the rightmost character entered is at the right boundary of the field. The Field Exit, Field+, or Field- (in a signed numeric field only) keys, when pressed, cause the data to shift. All character positions to the left of the characters entered are filled with 0's (bits 13-15 are set to 101 respectively) or with blank characters (bits 13-15 are set to 110 respectively).

Other rules that apply to an input field that is specified as a right-adjust field are:

- Signed numeric fields are always blank filled and right-adjusted unless otherwise specified.

- If no characters are entered in this field after the cursor is moved to this field, the Enter/Rec Adv key can be pressed.

- If any AID request key is pressed after some characters have been entered and the cursor is still in this field, a four-digit error code of 0020 is displayed.

DISPLAY STATION STATUS BYTES AND ERROR RECOVERY

When an error is sensed during a display station operation, bytes 10-15 of the input/output block (IOB) are posted with status indicating the cause of the error. Byte 13 of the input/output block is the AID byte (see Figure 7-3). If the AID byte is hex 04, a byte count error or an address translation error was sensed. However, if the AID byte is hex FF, the status information is posted in IOB bytes 10,11, 12, 14, and 15. (IOB bytes 10, 11, 12, 14, and 15 are equal to status bytes 0 through 4, respectively.)

The status posted in bytes 10 and 11 of the IOB specify the error conditions associated with the interface between the display station and the work station controller. The status posted in bytes 12, 14, and 15 of the IOB specify the error conditions associated only with the display stations.

Figure 7-13 describes the error conditions, gives the IOB bit definitions for each status byte, gives the priority in which the bits are tested, and gives a message number for each error condition. The message number is a reference to the *Displayed Messages Guide,* which contains the error recovery procedures needed for each of the error conditions.

Status Byte	Bit	Test Priority	IOB Byte 14 (in hex)	Bit Name	Bit Description (See Note)	Message Number (See Note)
0	0	7		Data stream reject	Invalid data was sensed in the output data stream while executing an output command. After this error occurs: 1. The processing of data ends immediately. Therefore, the states of the display screen and the format table are not predictable. 2. The error must be corrected before trying the same operation that caused the error. 3. A clear unit command should be executed to restore the display screen and format table to a known state. 4. Byte 14 of the IOB contains a code that gives the reason for the error. The following conditions cause this error:	
			01		● Premature termination of data stream: The byte count in the work station control field contains a number that causes the output data stream to end too soon.	SYS-7501 (SYS-6501)
			02		● Invalid address: An invalid address follows a set buffer address, repeat to address, or insert cursor order.	SYS-7502
			03		● Address is less than the current display address: The address specified in a repeat to address order is valid, but it is less than the current value in the display address register.	SYS-7503
			04		● Invalid output data stream: Either an escape character (hex 04) is missing, or the command byte is not valid. Valid command bytes include hex 01, 05, 11, 21, 23, 40, and 50.	SYS-7504
			05		● Invalid field length: The length bytes following the start of field order are 0.	SYS-7505

Figure 7-13 (Part 1 of 8). Display Station Status Bytes

Status Byte	Bit	Test Priority	IOB Byte 14 (in hex)	Bit Name	Bit Description (See Note)	Message Number (See Note)
0 (continued)	0		06	Data stream reject	• Invalid field starting address: An attempt was made to define an input field with a starting address that is less than, or equal to, the starting address of an input field already defined in the format table. The error is sensed when processing a start of field order.	SYS-7506
			07		• Incomplete or invalid data on a restore command: Either a restore command was sent to the wrong display station, or there was not enough information following the command (even if the command was sent to the correct display station).	SYS-7507
			08		• Input field too large: An attempt was made to define an input field that extends past the end of the display screen area. The error is sensed when processing a start of field order.	SYS-7508
			09		• Format table overflow: Too many input field are defined in the format table.	SYS-7509
			0A		• Display overrun: An attempt was made to write data past the end of the display screen area. This error is sensed when processing a write data command, or a write error command.	SYS-7510
			0B		• Invalid start of header parameter: The parameter byte following the start of header order is not hex 03.	SYS-7511
			0C		• Invalid roll parameter: One of the three parameter bytes following the roll command is not valid.	SYS-7512
			0D		• Too many field control words defined: Too many field control words are defined after a start of field order, or the work station controller sensed an invalid parameter byte following a start of field order (the first three bits of the byte are set to 000).	SYS-7513

Figure 7-13 (Part 2 of 8). Display Station Status Bytes

Status Byte	Bit	Test Priority	IOB Byte 14 (in hex)	Bit Name	Bit Description (See Note)	Message Number (See Note)
0	1	6		Work station control field error	Invalid data was sensed in the work station control field. After this error occurs: 1. The error must be corrected before trying the same operation that caused the error. 2. Byte 14 of the IOB contains a code that gives the reason for the error. If this error occurs, any data sent to the host should not be used. The following conditions cause this error:	
			01		● Invalid command modifier (IOB byte 4).	SYS-7533 (SYS-6533)
			02		● Invalid byte count (IOB bytes 8 and 9): The byte count is 0 for a display station, or it is more than 256 for a printer.	SYS-7534 (SYS-6534)
			03		● Invalid unit address (IOB byte 5).	SYS-7535 (SYS-6535)
					Note: The display screen and the format table are not changed by the above three errors, and no data is sent to (or received from) the display station.	
			04		● Incorrect byte count (IOB bytes 8 and 9): The byte count is not equal to the byte count associated with the following commands: – For the read screen, save screen, and save tables commands, the byte count is too small to permit the transfer of all data. – For the read input fields command, the byte count is either too small to permit the transfer of all data, or the byte count is not equal to 0 after all data has been transferred.	SYS-7536

Figure 7-13 (Part 3 of 8). Display Station Status Bytes

Status Byte	Bit	Test Priority	IOB Byte 14 (in hex)	Bit Name	Bit Description (See Note)	Message Number (See Note)
0	2	5		Resources temporarily not available	A requested operation cannot be executed at this time. The operation can be tried again when the condition causing the error is cleared. The following conditions cause this error:	
			02		● Work station is not operational	SYS-7566 (SYS-6566)
			03		● Work station is offline	SYS-7567 (SYS-6567)
			05		● Display station error state: The display station is in operator-error mode or system request mode, and a command cannot be executed. In operator-error mode, the only valid commands are save tables, save screen, restore tables, restore screen, write error (in response to the Help key), set-operator-alert-indicators, and reset-operator-alert-indicators. In system request mode, the only commands that are rejected are save tables, save screen, restore tables, and restore screen. *Note:* After this error occurs (and has been cleared), an AID byte of hex FA is generated. No other commands should be issued until the host receives the AID byte.	None
			06		● Read input fields command is rejected because the keyboard is not locked.	SYS-7570
			07		● Work station is not powered on: A command was sent to a work station that is not powered on (and mode set).	SYS-7571 (SYS-6571)

Figure 7-13 (Part 4 of 8). Display Station Status Bytes

Status Byte	Bit	Test Priority	IOB Byte 14 (in hex)	Bit Name	Bit Description (See Note)	Message Number (See Note)
0 (continued)	2		09	Resources temporarily not available	● Invalid save/restore command sequence: The command following a save tables or save screen command is not valid. Valid commands following: – A save tables command are save screen, set- (or reset-) operator-alert-indicators, clear unit, or another save tables command. – A restore tables command are restore screen, set- (or reset-) operator-alert-indicators, clear unit, or another restore tables command. *Note:* A save tables command and a restore tables command can be retried immediately, but a save screen command and a restore screen command cannot be retried unless preceded by a save tables or save screen command, respectively.	SYS-7573
0	3	3		Work station controller DBO/DBI parity check	A hardware error was sensed by the work station input/output control handler.	None
0	4	4		Operation check	A hardware error was sensed by the work station controller; operation of the work station controller ends until this error is corrected. The following conditions cause this error:	None
			01		● The SERDES failed; System/34 cannot communicate with the display stations.	
			02		● Cycle steal time-out occurred.	
			05		● Work station controller interrupts are not occurring. Therefore, keystrokes are not operational.	
0	5	1		Work station controller storage parity check	Even parity was sensed on the controller storage output.	None

Figure 7-13 (Part 5 of 8). Display Station Status Bytes

Status Byte	Bit	Test Priority	IOB Byte 14 (in hex)	Bit Name	Bit Description (See Note)	Message Number (See Note)
0	7	2		Work station controller long time-out check	Either a work station controller interrupt failed, or the work station controller microcode is in a loop.	None
1	0	17		Screen format error	An input field is not correctly defined; the error is most likely in the output data stream. Before operations can continue, the error must be corrected. The following conditions cause this error:	
			01		● An input field length error was sensed during a read input fields operation, or during keyboard entry into this field. – Field length was 0. – Field had no ending screen attribute. – Signed numeric field was 1 byte long. – Field read was not the same length as the field defined.	SYS-7601
			02		● Resequence error in the format table during a read input fields operation. – Resequence number is 0. – Resequence number is greater than the number of fields defined on the display screen.	SYS-7602
			03		● Check digit field was not valid during keystroke processing; the field was more than 32 characters long.	SYS-7603
1	1	9		No response time-out	Work station is no longer communicating with the work station controller. The work station is forced into hardware-error mode.	SYS-7300 (SYS-6300)
1	2	8		Transmit activity check	Work station controller attempted to communicate with a work station, but one of the following occurred: ● Communication was not established. ● Communication was established but with the wrong work station, or with more than one work station. The work station is forced into hardware-error mode.	SYS-7301 (SYS-6301)

Figure 7-13 (Part 6 of 8). Display Station Status Bytes

Status Byte	Bit	Test Priority	IOB Byte 14 (in hex)	Bit Name	Bit Description (See Note)	Message Number (See Note)
1	3	10		Activate command failure	The work station did not respond to an activate command, or the work station controller missed the response from the work station. The work station is forced into hardware-error mode.	SYS-7309 (SYS-6309)
1	4	11		Receive parity check	Data received by the work station controller from the work station was not valid. The work station is forced into hardware-error mode.	SYS-7303 (SYS-6303)
1	5	13		Receive length check	Work station controller received the wrong number of bytes from the work station. The work station is forced into hardware-error mode.	SYS-7306 (SYS-6306)
1	7	15		Even/odd response time-out	The work station did not switch its poll response level in 450 milliseconds. The work station is forced into hardware-error mode.	SYS-7390 (SYS-6390)
2	0	12		Device busy time-out	Work station did not execute a requested operation in the time allowed. The work station is forced into hardware-error mode.	SYS-7391 (SYS-6391)
2	1	16		Line parity check	Work station was not able to receive data from the work station controller. The work station is forced into hardware-error mode.	SYS-7304 (SYS-6304)
2	4-6	14		Exception status	Bits 4-6 are encoded as follows:	
					000 = No exception status	None
					001 = Null or attribute error for display station, or activate lost/discarded without a parity check for printer	SYS-7323 (SYS-6323)
					010 = Invalid activate	SYS-7324 (SYS-6324)
					011 = Reserved	SYS-7325 (SYS-6325)
					100 = Invalid command or device address	SYS-7320 (SYS-6320)
					101 = Input queue or storage overrun	SYS-7322 (SYS-6322)
					110 = Invalid register value for display station, or reserved for printer	SYS-7321 (SYS-6321)
					111 = Power on transition	SYS-7308 (SYS-6308)

Figure 7-13 (Part 7 of 8). **Display Station Status Bytes**

Status Byte	Bit	Test Priority	IOB Byte 14 (in hex)	Bit Name	Bit Description (See Note)	Message Number (See Note)
3	0-7	18		Status code 1	This byte contains hex 00 or one of the following: ● An error code (see bytes 0 and 1 in this chart). ● An invalid scan code if status byte 4 is equal to hex 2X. ● The high-order two digits of the operator-error code if the AID code in IOB byte 13 is equal to hex FB (Help key in operator-error mode).	None
4	0-7	18		Status code 2	This byte contains hex 00 or one of the following: ● Hex 28 if an invalid scan code is reported and the master modified data tag bit (bit 4 of this byte) is on. ● Hex 20 if an invalid scan code is reported and the master modified data tag bit is off. ● The low-order two digits of the operator-error code if the AID byte in IOB byte 13 is equal to hex FB (Help key from operator-error mode). ● Hex 08 if the master modified data bit is on.	None

Note: In the *Bit Description* column, the words *work station* refer to both display stations and printers. In the *Message Number* column, message numbers SYS-75XX are for the display stations and message numbers SYS-65XX are for the printers. The message numbers for the printer appear in this column only when the bit being described is common for both devices (display stations and printer).

Figure 7-13 (Part 8 of 8). Display Station Status Bytes

Chapter 8. Diskette

The diskette is used for two primary system functions:

- Storing data on the diskette to keep it for future use, then loading it back into the system when needed for a job
- Moving data, using the diskette for basic data exchange

The System/34 can have either a diskette 1 drive or a diskette 2D drive installed inside the 5340 system unit. The diskette 1 drive has one data head on one side of the diskette, for reading or writing. The diskette 1 drive can read from or write on only one side of a diskette in only frequency modulation recording mode. The diskette 2D drive has one data head on each side of the diskette, for reading or writing. The diskette 2D drive can read from or write on both sides of a diskette in either frequency modulation mode or modified frequency modulation mode.

The diskette drive turns the diskette at 360 ± 2.5% revolutions per minute. The diskette 1 drive can read or write 31,250 bytes per second. The diskette 2D drive can read or write either 31,250 bytes per second in frequency modulation mode or 62,500 bytes per second in modified frequency modulation mode.

DISKETTE SURFACE

The diskette surface is divided into cylinders. Each diskette surface contains 77 cylinders; cylinder 00 is the outside cylinder and cylinder 76 is the inside cylinder as shown in Figure 8-1.

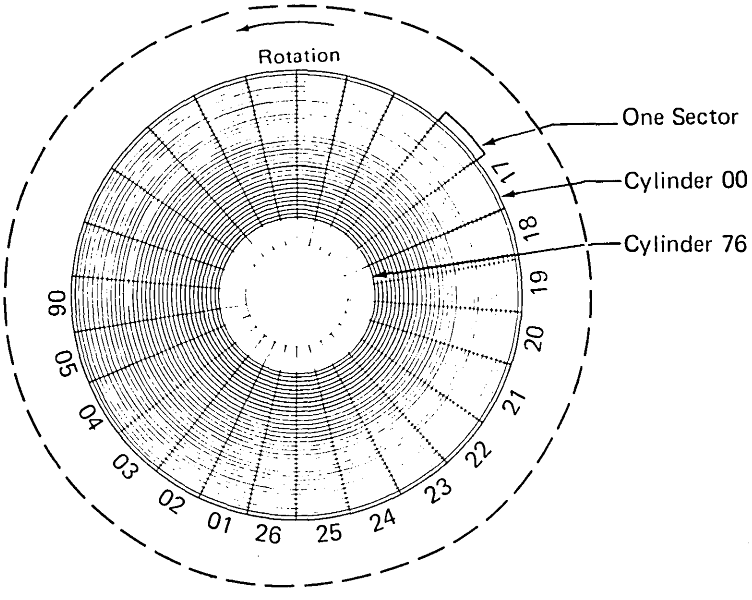


Figure 8-1. 26-Sector Diskette

Of the 77 cylinders, only 75 are normally used. Cylinder 00 is the index cylinder that contains the volume label and data set header labels; cylinders 1-74, the primary cylinders, store data records. Cylinders 75 and 76 are available for data storage in the event that one or two of the primary cylinders (1-74) becomes damaged.

If a damaged track is found during disk initialization, hexadecimal FF is written in the track sector identification fields. During a read operation, if an identification field of hexadecimal FF is read, the control storage program issues a seek to the next track and executes another read operation.

Cylinder 00 on a diskette 1, and cylinder 00 head 0 on a diskette 2, always have 26 sectors each having 128 bytes written in frequency modulation mode. Cylinder 00, head 1 on a diskette 2D always has 26 sectors, each having 256 bytes written in modified frequency modulation mode. Each cylinder has one track on a diskette 1, and two tracks on a diskette 2D.

For cylinders 1-74, each track is divided into either 8 or 26 sectors. In frequency modulation mode the 8 sectors each have 512-byte records and the 26 sectors each have 128-byte records. In modified frequency modulation mode the 8 sectors each have 1024-byte records and the 26 sectors each have 256-byte records.

SECTOR FORMAT

Each sector has a sector identification field, and has either a data record field or a control record field, as shown in Figure 8-2.

Therefore, because the diskette is formatted into cylinders and sectors, each record on the diskette has its own record address. This address is recorded at the physical location of the record on the diskette. Diskettes that contain record addresses are known as initialized diskettes.

DISKETTE INITIALIZATION

Diskettes are initialized in the Basic Data Exchange Format, as described in the *IBM Diskette General Information Manual*, GA21-9182.

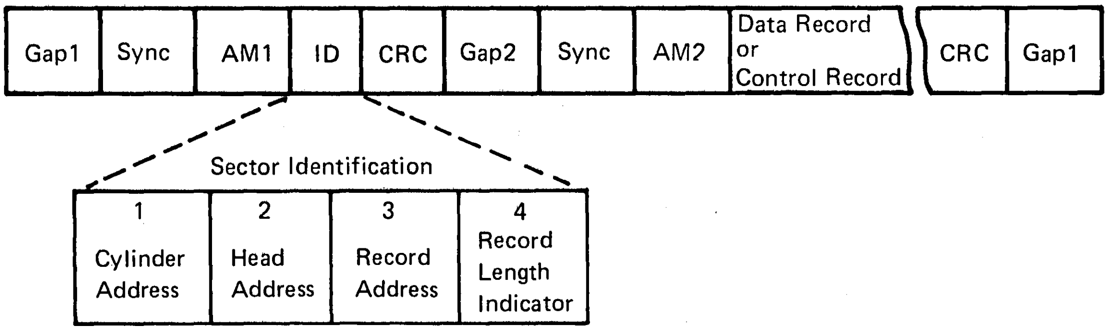


Figure 8-2 (Part 1 of 2). Sector Format

Field	Description
Gap 1	The gap between this record and the next record. It is variable length.
SYNC	A 6-byte field for FM mode or 12-byte field for MFM mode; all 0's; synchronizes the hardware circuits prior to reading the information from the diskette.
AM1	For FM mode a 1-byte identifier field address mark identifies the ID data and contains hex FE. For MFM mode a 4-byte identifier field address mark identifies the ID data and contains hex A1A1A1FE.
ID	A 4-byte sector identification (address) in the format CHRN, where: C = 1-byte binary cylinder address. Valid addresses are: Decimal = 00-76 Hex = 00-4C H = 1-byte binary head address. The valid address is X'00' or X'01'. R = 1-byte binary record address. Valid addresses are decimal 01-26 or 01-08, depending on the number of sectors per cylinder on this diskette. N = 1-byte record length indicator. N is hex 00 for 128-byte records, hex 01 for 256-byte records, hex 02 for 512-byte records, or hex 03 for 1024-byte records.
CRC	A 2-byte cyclic redundancy check field which verifies that the ID field and data field were read correctly. The system generates these bytes during a write operation and automatically performs a read-back check to verify their accuracy.
Gap 2	Gap between the ID field and the data field. It is generated by the system during write operations.
AM2	A 1-byte field for FM mode or 4-byte field for MFM mode. For FM mode the field contains either a hex FB or hex F8. Hex FB indicates the information following is a data field. Hex F8 indicates the information following is a control field. For MFM mode the field contains either a hex A1A1A1FB or A1A1A1F8. Hex A1A1A1FB indicates the information following is a data field. Hex A1A1A1F8 indicates the information following is a control field.
Data or Control	The length of this field is specified by the record length indicator (N) contained in the sector ID field. The contents of the control field are described in the *IBM Diskette General Information Manual*, GA21-9182.

Figure 8-2 (Part 2 of 2). Sector Format

DISKETTE OPERATIONS

The processing unit issues seek, read, and write operations to the diskette drive, which are controlled by the input/output block.

Diskette seek operations are overlapped with all other system functions. Diskette input/output operations are overlapped with all other system input/output functions except the disk storage drive functions.

Starting a Diskette Operation

A diskette operation is started by executing a supervisor call instruction with an R code of hexadecimal 41. Index register 1 must point to an assembled 34-byte input/output block.

Diskette Input/Output Block

The diskette input/output block controls diskette operations as shown in Figure 8-3.

If sequential sector addressing is used, bit 2 of the command modifier field of the input/output block must be a 0. If CHRNX addressing is used, bit 2 in the command modifier field of the input/output block must be a 1 and the CHRNX field (shown in the next paragraph) becomes bytes 30-34 of the input/output block.

00 *	01	02 *	03 *	04 *	05	06 *	07 *	08 *	09
Event Control Block	Completion Code	Flag Byte	Command Code	Command Modifier	Reserved (must be zero)	Data Address		Length (not used)	

0A	0B	0C	0D	0E	0F	10	11	12	13
Device Status Sense Byte 0	Device Status Sense Byte 1	Device Status Sense Byte 2	Device Status Sense Byte 3	Reserved (must be zero)	Reserved (must be zero)	TCB Pointer		Error Retry Count	Reserved (must be zero)

14	15	16 *	17 *	18 *	19	1A	1B	1C	1D
Reserved (must be zero)	Address of DTF		Next Sequential Sector to process		Sector Count	Sector Address		Number of Sectors to be Processed −1	Cylinder Address

*User Supplied Bytes

1E	1F	20	21
Head Address	Record Address	Record Length	Number of Sectors to be Processed −1

Figure 8-3 (Part 1 of 4). Diskette Input/Output Block

Displacement of Leftmost Byte in Hex	IBM Program Label	Length in Bytes	Field Description
0	$IOBRECB	1	Event control block *Bit* *Meaning When Set to 1* 0 Reserved. 1 The data address in the IOB is correct; do not translate it. 2–7 Reserved.
1	$IOBRCMP	1	This is the diskette completion code (in hex). 40 = Successful completion 41 = Permanent I/O error 42 = End of volume 43 = Not ready 44 = End of track 49 = Unsupported control record found
2	$IOBRFLG	1	Flag Byte *Bit* *Meaning When Set to 1* 0 No error recovery to be attempted. 1 Do not return on a permanent error. 2 Do not issue any error messages. 3 Do not log errors. 4 Disable automatic seek after a data operation. 5 Reserved, must be 0. 6 Reserved, must be 0. 7 Disable automatic seek before a data operation.

3	$IOBRCMD	1	Command Code
			D0 = Seek
			D1 = Read data (deleted control records are bypassed)
			D2 = Read data/control address mark (deleted control records are not bypassed)
			D3 = Read identification field
			D5 = Write data
			D6 = Write data/control address mark
			D7 = Write identification field

Figure 8-3 (Part 2 of 4). Diskette Input/Output Block

Displacement of Leftmost Byte in Hex	IBM Program Label	Length in Bytes	Field Description
4	$IOBRMDR	1	Command Modifier *Bit* *Meaning When Set to 1* 0 Data is in MFM recording mode. 1 Return on a not ready condition. 2 This IOB has a CHRNX field. 3 Control store data address (valid only for a read data command). 4 Diskette 2D. 5 Return control on end of track. 6,7 These bits define the physical sector length for sequential sector addressing: *Bit* *Sector Length* 0,0 128 bytes 0,1 256 bytes 1,0 512 bytes 1,1 1024 bytes
5	$IOBRSV1	1	Reserved, must be zero.
6	$IOBRDAT	2	Data address.
8	$IOBRLNG	2	Length (not used).
A	$IOBRSBO	1	Status byte 0. See Figure 8-4 for bit 0-4 meanings when status byte 1 bit zero is on.
B	$IOBRSB1	1	Status byte 1.
C	$IOBRSB2	1	Status byte 2.
D	$IOBRSB3	1	Status byte 3.
E	$IOBRSV2	1	Reserved, must be zero.

Figure 8-3 (Part 3 of 4). Diskette Input/Output Block

Displacement of Leftmost Byte in Hex	IBM Program Label	Length in Bytes	Field Description
F	$IOBRSV3	1	Reserved, must be zero.
10	$IOBRTCB	2	These 2 bytes contain the address of the task control block associated with this input/output block. There is one task control block for each task that can be performed by the system. More than one task can be executing at the same time on the system. The number of tasks that can be run on a system depends on the system configuration.
12	$IOBRERR	1	Error retry count.
13	$IOBRPAD	2	Reserved—must be zero.
15	$IOBRDTF	2	Define the file address.
17	$IOBRLSP	2	Next sequential sector to process. If SS addressing is used (bit 2 of the command modifier is 0), and a read or write operation has ended without an error, this field contains the SS of the last sector processed, +1. If there is an error, this field contains the SS of the sector in error.
19	$IOBRLFT	1	Residual sector count. It is either a hex FF which indicates that all data sectors were transferred or it is the number of sectors that were not transferred, -1.
1A	$IOBRSS	2	Starting sector address. If bit 2 of the command modifier is a 1, and a read or write operation is complete, this field contains the final cylinder and head of the last operation.
1C	$IOBRNB	1	Number of sectors of data transferred, –1. If bit 2 of the command modifier is a 1, upon completing a read or write operation this field contains the last sector number transferred, +1. The following fields are used only if bit 2 of the command modifier is a 1.
1D	$IOBRCYL	1	Cylinder (C).
1E	$IOBRHD	1	Head (H).
1F	$IOBRRCD	1	Record number (R).
20	$IOBRSIZ	1	Sector size (N). Bits 6 and 7 define the physical record length. These values are the same as bits 6 and 7 of the command modifier.
21	$IOBRNUM	1	Number of sectors (x) of data transferred, -1.

Figure 8-3 (Part 4 of 4). Diskette Input/Output Block

CHRNX Field

The 5-byte CHRNX field specifies:

- For a seek operation, C is either a 00 through 76 for the desired seek track address, or hexadecimal FF which specifies a recalibrate operation. H, R, N, and X are not used for a seek operation.

- For a read data operation, a read control record operation, a write data operation, or a write control record operation:
  - *C (cylinder number)* is a 1-byte address; valid decimal addresses are 0 through 76.
  - *H (head number)* is (1) for a diskette 1, a 1-byte address of 0 or (2) for a diskette 2D, either a 0 for data head zero or a 1 for data head one.
  - *R (record number)* is a 1-byte address that specifies the first record to be processed in a one-record or many-record data operation. Decimal record numbers 1 through 26 are valid for 128-byte frequency modulation or 256-byte modified frequency modulation format diskettes. Record numbers 1 through 8 are valid for 512-byte frequency modulation or 1024-byte modified frequency modulation format diskettes. Not valid record numbers set the record not compare status bit (status byte 0, bit 5).
  - *N (record length indicator)* is a 1-byte binary number that indicates the physical record length. N must be 0 for 128-byte records, 1 for 256-byte records, 2 for 512-byte records, and 3 for 1024-byte records. This byte does not change as the operation is executed.
  - *X (number of records)* is a 1-byte binary number that specifies the number of records to be processed minus 1.

Before processing any data field, the system searches the track for the desired sector identification field (CHRN). This verifies that the seek operation found the correct cylinder. If the fields do not compare, the system indicates the error in the status bytes at the input/output block.

Diskette Addressing

Sequential Sector Address

If actual sector (CHRNX) addressing is not used to identify a diskette data area, sequential sector addressing is used. Sequential sector addressing starts at hexadecimal address 0001 (cylinder 1, data head 0, sector 1), increases by 1 for each following sector, and extends through the last sector on cylinder 74. Cylinder 0 cannot be addressed by sequential sector addressing.

The control storage program changes sequential sector addresses to actual sector addresses for all operations.

Actual Sector Address

If sequential sector addressing is not used, actual sector addressing is used. The user supplies the 5-byte CHRNX field in bytes 30 through 34 of the input/output block. This method of addressing must be used to address cylinder 0 (the volume label and volume table of contents).

Diskette Seek Mechanism

Seek

The diskette drive seek mechanism moves the data head to the cylinder specified by the sequential sector address or CHRNX. A seek is performed on all data operations except a write identification and read identification if:

- The data head is not already at the correct cylinder address.

- Bit 7 of the input/output block flag byte is zero.

A seek to the next cylinder is performed after a write data operation, or a read data operation if:

- The last sector of a cylinder has been processed.

- Bit 4 of the input/output block flag byte is zero.

Recalibrate

The diskette drive seek mechanism moves the data head to cylinder 00. A recalibrate is the only way to clear a not ready condition. The recalibrate operation is started by a seek operation and either a sequential sector address of hexadecimal FFFF or the cylinder byte of CHRNX being hexadecimal FF as specified in the input/output block.

Read Operations

Read Data

This operation reads data starting at the data record specified by the sequential sector address or the CHRNX address. The system reads X+1 data records into main storage. (X is either byte 19 or byte 1E of the input/output block.) Deleted control records are not read into main storage.

Read Control Record

This operation is the same as read data, except that deleted control records are read into main storage.

Read Identification

This operation reads one 4-byte sector identification field (CHRN) from the track now under the data head, and sends it to main storage.

Write Operations

An automatic write verify (which cannot be disabled) is performed during all write operations. The verify is done by reading the data just written and comparing that data against the data in the data area in main storage.

Write Data

This operation sends data records from the data field in main storage, and the record is written in the specified diskette sector data record field. This operation continues until data from main storage has been written to X+1 diskette data sectors. (X is the contents of either byte 1C or byte 21 of the input/output block.)

Write Control Record

This operation sends deleted control records from the data field in main storage, and the record is written in the specified sector control record field. The first byte in the data area in main storage must be a character D (delete record). Records written this way will not be read by a read data operation. The first control record of data in the data area in main storage is written in X+1 diskette sectors. (X is the contents of either byte 1C or byte 21 of the input/output block.)

Write Identification

This operation formats the track that is now under the data head and new sector identification fields and data fields are written. The first data field in main storage is written in all data fields of the track.

CHECK CONDITIONS AND STATUS

For a summary of diskette operation-ending conditions, diskette indicators set, and restart procedures see Figure 8-5. Except where differently indicated, all status bytes are reset before executing an operation.

Status Byte 0

Note: For the other meaning of the following four bits (0-3) if the no-operation bit is on (status byte 1, bit 0) see Figure 8-4.

Bit Description

0 *Missing data address mark*: Data was not found after a sector identification field.

1 *Identification cyclic redundancy check*: Indicates a cyclic redundancy check in a sector identification field.

2 *Data cyclic redundancy check*: Indicates a cyclic redundancy check in the data field.

3 *Cylinder not compare*: The cylinder address part of the sector identification field and the desired cylinder byte did not match during a sector identification search.

4 *Head not compare*: The data head address part of the sector identification field and the desired data head byte did not match during a sector identification search.

5 *Record not compare*: The record address part of any sector identification field and the desired record byte did not match during a sector identification search.

6 *Length not compare*: The record length part of the sector identification field and the desired length did not match during a sector identification search.

7 *Seek reverse*: The last seek was in the reverse direction.

Status Byte 1

Bit Description

0 *No operation*: A diskette operation cannot be executed because of existing status before executing a diskette operation. This bit is reset by the next diskette operation or by a system reset.

Note: If this bit is on, see Figure 8-4 for other meaning of status byte 0 bits 0-3.

1 *Control record not valid*: The leftmost byte of a control record contains other than a F or D, control address mark.

2 *Verify compare error*: The data written does not match the main storage data field.

3 *Control address mark record found*: A control address mark was found while performing a Read Data or a Read Data and Control Record operation.

4 *Not used*

5 *Write error*: Either a write overrun (status byte 2, bit 6), a write parity check (status byte 2, bit 7), a missing erase current (status byte 2, bit 2), data unsafe (status byte 3, bit 0) was indicated during a write operation.

6 *End of track*: The last record on the track has been processed, but some records remain to be processed.

7 *File busy*: Data is being read from or written on the diskette. Interrupt Level 1 is being used.

Status Byte 0	Description
Bit 0 1 2 3	
0 0 0 0	Device address or port address not valid: The control storage program has detected that the diskette device address or port address received from the IOB is not valid.
0 0 0 1	Command not valid: The control storage program has detected that a diskette command. received from the IOB is not valid.
0 0 1 0	Not ready—not seek operation: The command issued to the diskette is rejected because the diskette is not ready and the command is not a seek.
0 0 1 1	Not ready—seek is not a recalibrate: The command issued to the diskette is rejected because the diskette was not ready and the command was not a recalibrate.
0 1 0 0	Errors not reset: One or more errors have not been reset.
0 1 0 1	Reject head 1 operation: The command issued to the diskette is rejected because head 1 is selected, but a diskette 1 is in the machine.
0 1 1 0	Reject MFM operation: The MFM command issued to the diskette is rejected because a diskette 1 drive is in the machine.
0 1 1 1	Write gate or erase gate on: A command should not be issued to the diskette because the write gate or erase gate is not off.

Figure 8-4. No Operation Status

Status Byte 2

Bit Description

0 *Diskette over speed*: The diskette is turning quicker than the maximum specified rate of 376 revolutions per minute, or 159 milliseconds per revolution.

1 *Diskette not ready*: indicates that:

- The diskette is not inserted, or
- The diskette cover has been opened and closed, or
- The diskette is not up to the correct speed, or
- The diskette is inserted backward.

2 *Erase current missing*: Erase current failed to turn on during a write operation.

Note: This status bit is reset by the verify part of a write operation. Write error status (status byte 1, bit 5) is set at the same time; the program can sense this bit.

3 *Record not found*: The CHRN sector identification field could not be found in the selected track during a sector identification search.

4 *Read overrun*: The minimum data rate from the diskette to main storage was not maintained.

5 *Data mode*: Off only during a modified frequency modulation read or write operation.

6 *Write overrun*: The minimum data rate from main storage to the diskette was not maintained.

Note: This status bit is reset by the verify part of a write operation. The write error status (status byte 1, bit 5) is set at the same time; the program can sense this bit.

7 *Write parity check*: The data bus out parity and system generated serial write data parity did not match during a write operation.

Note: This status bit is reset by the verify part of a write operation. The write error status (status byte 1, bit 5) is set at the same time; the program can sense this bit.

Status Byte 3

Bit Description

0 *Data unsafe*: Erase current is on while not in a write operation.

Note: This status bit is reset by the verify part of a write operation. Write error status (status byte 1, bit 5) is set at the same time; the program can sense this bit.

1 *Block processor clock off*: Diagnostic use only.

2 *One head drive*: A diskette 1 drive is installed.

3 *Erase current off*: Diagnostic use only.

4 *Head 0 active*: Diagnostic use only.

5 *One-surface diskette*: Use only data head 0.

6 *I/O working off*: No device is busy.

7 *Diskette working off*: The diskette is not busy.

Condition Set By:

Status Byte 0

Bit	Description	Seek	Read Data	Read ID	Read Data Control	Write Data	Write Control	Write ID	Suggested Action[1]
0	Missing data address mark		X		X	X	X	X	3
1	Identification cyclic redundancy check		X	X	X	X	X	X	3
2	Data cyclic redundancy check		X		X	X	X	X	3
3	Cylinder not compare		X		X	X	X	X	2
4	Head not compare		X		X	X	X	X	3
5	Record not compare		X		X	X	X	X	3
6	Length not compare		X		X	X	X	X	3
7	Seek reverse								

Status Byte 1

Bit	Description	Seek	Read Data	Read ID	Read Data Control	Write Data	Write Control	Write ID	Suggested Action[1]
0	No operation	X	X	X	X	X	X	X	3
1	Control record not valid		X				X		4
2	Verify compare error								
3	Control address mark record found								
4	Not used								
5	Write error					X	X	X	3
6	End of track		X		X	X	X		2
7	File busy								

[1] Actions are described in Part 3 of this figure.

Figure 8-5 (Part 1 of 3). Diskette Check Conditions and Status

Status Byte 2

Condition Set By:

Bit	Description	Seek	Read Data	Read ID	Read Data Control	Write Data	Write Control	Write ID	Suggested Action[1]
0	Diskette over speed	X	X	X	X	X	X	X	3
1	Diskette not ready	X	X	X	X	X	X	X	1
2	Missing erase current					X	X	X	3
3	Record not found		X	X	X	X	X	X	3
4	Read overrun		X	X	X	X	X	X	3
5	FM mode								
6	Write overrun								
7	Write parity check								

Status Byte 3

Bit	Description	Seek	Read Data	Read ID	Read Data Control	Write Data	Write Control	Write ID	Suggested Action[1]
0	Data unsafe	X	X	X	X	X	X	X	3
1	Block processor clock off								
2	Two head drive								
3	Erase current off								
4	Head 0 active								
5	One surface disk								
6	I/O working off								
7	Diskette working off								

[1] Actions are described in Part 3 of this figure.

Figure 8-5 (Part 2 of 3). Diskette Check Conditions and Status

Suggested Recovery Action	
Number	**Description**
1	If this is the second try, then recalibrate and verify that the correct diskette is in the diskette drive. If this is the third try, then exit to a permanent error routine requiring operator intervention.
2	Issue a seek to the logical cylinder desired, then issue the data operation again. Try three or more times. If not successful, post a permanent error.
3	a. Try original operation or sequence of operations a maximum of three times if a write operation. Try 10 times if a read or seek operation. b. If try is successful, then return to processing. c. If try is not successful, call a permanent error log out routine, then return to processing.
4	Call a not-valid-control-record routine and return to processing.

Figure 8-5 (Part 3 of 3). Diskette Check Conditions and Status

ERROR RECOVERY

The recovery procedures shown in Figure 8-5 can be used to restart the system operation. The type of action necessary is determined by performing a sense diskette operation, then testing to determine which diskette status bits are on.

INITIAL PROGRAM LOAD

The source of control storage initial program load and main storage initial program load is selected by the CSIPL and MSIPL switches on the CE panel. Pressing the Load key on the operator panel starts control storage initial program load.

If the CSIPL switch is set to the Diskette position, the control storage program is loaded from the diskette. If the CSIPL switch is set to the Disk position, the control storage program is loaded from the disk.

A diskette control storage initial program load sends a 4096-byte diagnostic program located on track 00, record 01, to control storage. This program is used by the system, not by the programmer. After this diagnostic program has executed, the control storage system control program is loaded. This program tests the MSIPL switch setting.

If the MSIPL switch is set to the Diskette position, the main storage initial program is loaded from the diskette. If the MSIPL switch is set to the Disk position, the main storage program is loaded from disk. The main storage initial program load from the diskette is usually used for installing a new supervisor or changing the system configuration.

Chapter 9. Data Communications

Data communications (BSC or SDLC) is a special feature of System/34; it lets the system function as a point-to-point or multipoint terminal. Operation is half-duplex, synchronous, and serial by bit, serial by character over either switched voice-grade two-wire lines, or nonswitched two- or four-wire lines.

Operation of the communications feature is controlled by System/34 stored program instructions and, for BSC, by responses to line-control characters.

DATA COMMUNICATION NETWORKS

Point-to-Point Networks

Data communications functions on either a switched or nonswitched point-to-point network. On a nonswitched network, data transmissions are always between the same two stations. On a switched network, data transmissions between any two stations on the network are made by dialing.

Multipoint Networks

All stations on a multipoint network are permanently connected (nonswitched) and all data transmissions are between two stations: the control station and an addressed tributary station. System/34 is used on a multipoint network as a tributary station; it cannot be used as a control station.

TRANSMISSION DATA RATES

A clock controls the rate at which data is transmitted and received. Either a special feature (see *Internal Clock*, later in this chapter), the network, or the modem supplies the clock. However, connected stations must use the same clocking source.

The communications feature can operate at various data rates between 600 and 9,600 bps (bits per second). The data rate used is determined by the modem, and connected stations must operate at the same data rate using compatible modems.

SPECIAL COMMUNICATIONS FEATURES

The following special communications features are available on System/34.

EIA/CCITT Interface

The EIA/CCITT (Electronic Industries Association/International Telephone and Telegraph Consultative Committee) interface feature supplies an interface adapter and cable for attaching the communications adapter to an external modem. The interface cannot be installed with the IBM 1200 BPS Integrated Modem, the IBM 2400 BPS Integrated Modem, or the Data-Phone[1] digital service adapter, all of which are described in this chapter. The interface might need the internal clock feature (described in the following paragraph) if the external modem does not supply its own clocking.

Internal Clock

The internal clock supplies transmission rates of 600 bps and 1,200 bps, and supplies a clocking system in the communications adapter to permit operation with modems that do not supply clocking to the adapter. The internal clock can be installed with the EIA/CCITT interface and must be installed with the IBM 1200 BPS Integrated Modem; it cannot be installed with the IBM 2400 BPS Integrated Modem or the Data-Phone digital service adapter.

Data-Phone Digital Service Adapter

The Data-Phone digital service adapter supplies an interface for attaching the communications adapter to AT&T's digital data network. The transmission rates for this adapter are 2400, 4800, or 9600 bps.

[1] Trademark of the American Telephone & Telegraph Co.

Data is transmitted serially-by-bit and serially-by-character to the digital data network. A clock is supplied by the digital data network for clocking the data to and from the adapter.

Note: When the Data-Phone adapter is installed, up to 1 second of interference is transmitted to the network when the system's power is turned on or off.

STANDARD COMMUNICATIONS ADAPTER FEATURES

The following two features, rate select and automatic answering, are supplied with every communications adapter operating on a switched network.

Rate Select

Rate select permits programs to transmit at half the normal speed if the system has a modem that can operate at half rate.

Automatic Answering

Automatic answering (switched network only) enables the communications adapter to respond to a telephone request for data communications automatically without operator action (manual answer) if the modem also has the automatic answering feature.

MODEMS

The modem receives the data serially by bit and serially by character from the communications line during receive operations and sends the bits to the communications adapter. During transmit operations the communications adapter receives characters in parallel from storage, then makes them available serially by bit, serially by character to the modem. The modem, in turn, places each bit on the communications line as soon as it receives the bit from the communications adapter.

IBM 1200 BPS Integrated Modem

This modem permits communications at a data transmission rate of 1200 bits per second over a nonswitched or switched network. The device communicating with System/34 must also have an IBM 1200 BPS Integrated Modem, or a compatible modem.

This modem comes in two types:

- The *nonswitched type* attaches to two- or four-wire lines through an IBM-supplied cable directly to the line.

- The *switched with automatic answering type* attaches to a switched network through an IBM-supplied cable to a common carrier data access arrangement (CBS type coupler or similar coupler). The IBM 1200 BPS Integrated Modem needs the internal clock. It cannot be installed with the EIA/CCITT interface, the IBM 2400 BPS Integrated Modem, or the Data-Phone digital service adapter.

IBM 2400 BPS Integrated Modem

This modem permits communications at a data transmission rate of 2400 bits per second over a nonswitched or switched network. The device communicating with System/34 must also have an IBM 2400 BPS Integrated Modem or an IBM 3872 Modem. The IBM 2400 BPS Integrated Modem cannot be installed with the EIA/CCITT interface, the IBM 1200 BPS Integrated Modem, or the Data-Phone digital service adapter.

This modem comes in the following types:

- The *nonswitched point-to-point and nonswitched multipoint types* attach to two- or four-wire lines through an IBM-supplied cable directly to the line.

- The *switched, switched network backup, and the switched with automatic answering types* attach to a switched network through an IBM-supplied cable to a common carrier data access arrangement (CBS type coupler or similar coupler).

Modem Features for the IBM 2400 BPS Integrated Modem

Two features that can be installed with the IBM 2400 BPS Integrated Modem (nonswitched) are the switched network backup feature and the switched network backup with the automatic answering feature. These features are described in the following paragraphs.

Switched Network Backup (SNBU)

This feature permits backup attachment of System/34 to the switched network if the primary modem is the IBM 2400 BPS Integrated Modem on a nonswitched line. Communication can be with either another IBM 2400 BPS Integrated Modem or an IBM 3872 Modem if these modems have the switched or the switched network backup feature. Switched network backup cannot be installed with the switched network backup with automatic answering feature, which is described in the following paragraph. Attachment to the switched network is through an IBM-supplied cable to the common carrier data access arrangement (CDT type coupler or similar coupler).

Switched Network Backup with Automatic Answering

This feature is the same as the switched network backup feature, described in the preceding paragraphs, except that it automatically answers incoming calls when attached to a common carrier data access arrangement (CBS type coupler or similar coupler). It cannot be installed with the switched network backup feature.

BSC ADAPTER

This section of Chapter 9 pertains to the BSC adapter only; for a description of the SDLC adapter, see *SDLC Adapter* later in this chapter.

Transmission Codes

Data can be transmitted in either of two codes, EBCDIC (extended binary-coded decimal interchange code) or ASCII (American National Standard Code for Information Interchange). In each job that uses the BSC adapter, the customer must specify once which code is being used in the job. Only stations using the same code can communicate with each other.

EBCDIC and ASCII have different codes to represent characters. EBCDIC is the standard, 8-bit plus parity, internal binary code of System/34. The bits are numbered 0 through 7 starting at the high-order bit. The parity bit, used for internal checking, is not transmitted over the communications network.

ASCII is a 7-bit plus parity code. In ASCII, the bits are numbered 1 through 7 starting at the low-order bit (Figure 9-1).

All characters are transmitted over the line low-order bit first. For ASCII, the high-order bit must be a zero bit from main storage on transmit. If the adapter does not receive a high-order zero from main storage, it generates and sends out a wrong-parity (P) bit. In addition, the invalid-ASCII-character status bit is set on, causing a unit check condition.

	First Hex	Second Hex
	High	Low
Order of 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

Note: The complete EBCDIC character set and the ASCII character set are shown in Appendix F.

Figure 9-1. EBCDIC and ASCII Bit Positions

BSC Adapter Features

The following features are standard with the BSC adapter for System/34.

Transmission Code Selection. The BSC adapter can transmit and receive both EBCDIC and ASCII data. (Only stations using the same transmission code can communicate with each other.) The transmission code used can be changed for each job.

Intermediate Block Checking. Intermediate block checking permits intermediate text block (ITB) characters to be received for checking the accuracy of communication without interrupting the constant flow of information from the transmitting station to the receiving station.

Full Transparent Text Mode. Full transparent text mode (EBCDIC only) permits any of the 256 EBCDIC bit combinations to be transmitted as data. Therefore, the EBCDIC line-control characters can, if needed, be transmitted as data.

BSC Input/Output Block

Program operation of BSC is controlled by an input/output block (IOB) issued by the SVC I/O request instruction. The IOB, as shown in Figure 9-2, contains all information needed to carry out a requested operation. The SVC I/O request queues the operation to the adapter. (For BSC, only one operation may be queued at a time.) When the operation ends, the IOB contains the status of the operation. The IOB must be posted complete before the next SVC I/O request for BSC is issued.

Posting IOBs Complete

IOBs are posted complete at the end of the following BSC operations:

- Receive only (Q code = hex 81)

- Receive initial delayed (Q code = hex 82)

- Receive initial (Q code = hex 83)

- Transmit/receive overlay (Q code = hex 84)

- Transmit/receive initial (Q code = hex 85)

- Transmit/receive (Q code = hex 86)

- 2-second time-out. (The BSC adapter need not be enabled to complete the 2-second time-out operation.)

- Enable or disable.

On a receive operation, an IOB is posted complete when a changeof-direction character is decoded, when the receive buffer is filled, or when a receive time-out occurs. On a 2-second time-out operation, an IOB is posted complete at the end of the 2-second time-out, or when an I/O request instruction is issued.

Displacement of Leftmost Byte in Hex	IBM Program Label	Length in Bytes	Description
0	IOBECM	1	Event control mask *Hex Meaning* 80 Do not skip indicator. 40 Data buffer address is real.
1	IOBHCMP	1	Completion code *Hex Meaning* 80 Active: Set on by control storage when processing the IOB and set off by control storage when processing is complete. If this bit is on, processing of the IOB is not permitted. 40 Complete: Set on by control storage when the IOB is complete; set off by main storage. If this bit is on, processing of the IOB is not permitted. 02 Forced 2-second time-out. 01 Error detected: Set on if any bit in status byte 0 is on.
2	IOBPARM	1	Parameter byte
3	IOBQ	1	Command (Q) code Bits 0 1 2 3 = Attachment address Bits 4 5 6 7 = Command type 0 0 0 0 Control. 0 0 0 1 Receive only. 0 0 1 0 Receive initial delayed. 0 0 1 1 Receive initial. 0 1 0 0 Transmit/receive overlay: Same as the transmit/receive command, but the received data overlays the transmit buffer. 0 1 0 1 Transmit/receive initial: The receive part of the buffer must follow, and be contiguous to, the transmit part of the buffer. 0 1 1 0 Transmit/receive: The receive part of the buffer must follow, and be contiguous to, the transmit part of the buffer. This command is the same as the transmit/receive initial command, but this command does not start until the microcode recognizes the EOT character. 0 1 1 1 Reserved. *Note:* The BSC control storage program does not check the validity of the command code. The program decodes the command from only the low-order 3 bits and proceeds with the operation. Where the low-order 3 bits are defined to be reserved, the operation defaults to a receive initial operation.

Figure 9-2 (Part 1 of 3). BSC IOB

Displacement of Leftmost Byte in Hex	IBM Program Label	Length in Bytes	Description
4	IOBR	1	Command modifier When the command (Q) code is hex 0: C0 = Enable BSC 80 = Disable BSC 04 = Start 2-second timer 01 = Microcode looks for poll or address without preceding EOT When command (Q) code is not hex 0, the command modifier bits are reserved.
5	IOBADR	1	Station address (multipoint tributary)
6	IOBSTAR	2	Data buffer address: Points to the start of the data buffer.
8	IOBRLN	2	Data buffer length (receive): Defines the number of bytes in the receive portion of the data buffer.
A	IOBSNS1	1	Status byte 1 *Hex Meaning* 80 Receive time-out 40 Block check 20 Transmit adapter check 10 Receive adapter check 08 Invalid ASCII character 04 Abortive disconnect 02 Not data set ready 01 Reserved
B	IOBSNS2	1	Status byte 2 (reserved)
C	IOBCAR	2	Buffer address: Indicates the last position of the buffer (plus 1 position) that was used at the completion of a receive, receive initial, transmit/receive, transmit/receive initial, or a transmit initial delayed command.
E	IOBRES	2	Reserved
10	IOBTCB	2	Task control block address
12	IOBQHDR	1	Queue identification

Figure 9-2 (Part 2 of 3). BSC IOB

Displacement of Leftmost Byte in Hex	IBM Program Label	Length in Bytes	Description
13	IOBLDEF	1	Line definition byte *Hex* *Meaning* 80 Half rate is selected. 40 Internal clock is needed. 20 IBM modem is installed. 10 Answer tone must be supplied by data terminal equipment. 08 Standby line is selected. 04 Multipoint line is selected. 02 Switched data terminal ready line is selected. 01 Nonswitched line is selected.
14	IOBTLN	2	Data buffer length (transmit): Defines the number of bytes in the transmit portion of the data buffer.
16	IOBDUM2	2	Reserved

Figure 9-2 (Part 3 of 3). BSC IOB

BSC Adapter Controls

A station controls its BSC adapter with control, pad, and synchronization characters. These characters are described in the following paragraphs, and they include:

- Starting codes, to enter specific modes and to start accumulation of BCCs (block check characters; see *Data Checking and BSC Status Bytes*, later in this chapter)
- Modifiers, synchronization characters, and data link escape functions (ITB, SYN, DLE)
- Ending codes, to terminate blocks and activate checking functions

Control Characters

The BSC control characters are described in the following paragraphs and are shown in Figure 9-3.

- *SOH (start of header) or STX (start of text)* precede a block of text characters. Both reset control mode and set the adapter to data mode. The first SOH or STX after line turnaround resets the BCC buffer, and BCC accumulation starts with the following character.
- *ETB (end of text block) or ETX (end of text)* terminate a block of characters started with SOH or STX. Both ETB and ETX reset data mode in the adapter and are the last character included in the BCC accumulation. At the primary station, the adapter transmits the BCC and the pad character. At the secondary station, the adapter compares its BCC accumulation with the BCC(s) received following the ETB or ETX.
- *EOT (end of transmission)* indicates the end of a transmission, which may contain more than one message, and resets all stations on the line to control mode. EOT is also transmitted as a negative response to a polling sequence. EOT cannot be immediately preceded by any character other than SYN. To be recognized as a control character, EOT must be followed by four consecutive binary 1's.

- *ENQ (enquiry)* resets data mode in the adapter.

- *NAK (negative acknowledgment)* indicates that the preceding transmission block was in error and the receiving station is ready for another transmission of the same block. NAK is also the *not ready* response to multipoint station selection sequences and point-to-point initialization sequences. NAK must be followed by four consecutive binary 1's to be recognized as a control character.

- *SYN (synchronous idle)* is generated and transmitted automatically by the adapter to establish and maintain synchronization. A SYN from main storage at the transmitting station is transmitted, but does not enter main storage at the receiving station or BCC accumulation at either station.

- *SYN SYN* is the synchronization pattern in nontransparent mode. Two consecutive SYN characters are always transmitted immediately after an ITB or a BCC sequence.

- *DLE (data link escape)* informs the adapter to test the following character for a control sequence. In nontransparent text mode, DLE is data.

- *ITB (intermediate text block)* is included in the BCC and causes the BCC(s) to be sent or compared. Both adapters continue in data mode with the new BCC accumulation starting with the first non-SYN character.

- *ACK 0 (even acknowledgment) and ACK 1 (odd acknowledgment)* are positive acknowledgments by the receiving station that the preceding even-numbered (ACK 0) or odd-numbered (ACK 1) transmission block was received. In data mode, ACK indicates that the last block check character received matched the block check character generated by the adapter. In control mode, ACK indicates that the adapter is ready to receive. ACK always needs a response from the station that receives it. ACK causes the receiving adapter to end the receive operation and post the IOB complete.

- *WACK (wait before transmit—positive acknowledgment)* signals that the last data block was received correctly but the receiving station cannot continue receiving. During line initialization, a received WACK indicates that the remote station cannot receive any data immediately but can receive data in a short time. WACK causes the receiving adapter to end the receive operation and post the IOB complete.

- *DISC (mandatory disconnect)* is transmitted (in switched point-to-point networks only) to signal the remote station that the transmitting station is going to disconnect from the line. DISC causes the receiving adapter to end the receive operation and post the IOB complete.

- *RVI (reverse interrupt)* is transmitted by a secondary station to request that the primary (control) station end its transmission and permit the secondary station to transmit. RVI is transmitted in place of ACK. Sequential RVIs can be transmitted only in response to ENQ. RVI causes the receiving adapter to end the receive operation and post the IOB complete.

  In addition, on a multipoint network, RVI is transmitted by the tributary station as an acknowledgment to a select sequence, and as an indication that the tributary station wants to transmit.

- *TTD (temporary text delay)* is transmitted by a primary station to inform the secondary station that (1) there will be a delay of more than two seconds in transmitting the next data block, or (2) the primary station wants to cancel the transmission. The secondary station responds to TTD by transmitting NAK. TTD causes the receiving adapter to end the receive operation and post the IOB complete.

- *XSTX (transparent start of text)* resets control mode and sets the adapter to data mode and transparent mode. Unless preceded by SOH—, XSTX resets the BCC register and BCC accumulation starts 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 a DLE for each DLE received from main storage.

- *XITB (transparent intermediate block)* causes the same adapter action as ITB and, in addition, resets transparent mode.

- *XETX or XETB (transparent end of text or transparent end of text block)* causes the same adapter action as ETX or ETB and, in addition, resets transparent mode.

- *XSYN (transparent synchronous idle)* is the synchronization pattern for transparent mode. It does not enter BCC or main storage.

- *XENQ (transparent block cancel)* resets data mode and transparent mode in the adapter.

- *XTTD (transparent TTD)* performs the function of TTD in transparent mode.

- *XDLE (transparent DLE)* is interpreted in transparent mode as a valid data byte (hexadecimal 10).

Pad Characters for BSC

The BSC adapter generates and sends one pad character for each change-of-direction character transmitted. If the change-of-direction sequence calls for a block check character, the pad character follows the block check character; if not, the pad character follows the change-of-direction character in the message being transmitted. This pad character is hexadecimal FF.

The BSC adapter also generates and sends a pad character as the second character of the NAK and EOT control character sequences.

When transmission starts, the BSC adapter automatically generates and inserts a pad character (in this case, a hexadecimal 55) in front of the first synchronization sequence. No leading or trailing pad character (except a pad character immediately following either EOT or NAK) is stored during receive operations.

Name	Mnemonic	EBCDIC	ASCII
Start of header	SOH	SOH	SOH
Start of text	STX	STX	STX
End of text 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 acknowledgment	NAK	NAK	NAK
Synchronous idle	SYN	SYN	SYN
Data link escape	DLE	DLE	DLE
Intermediate block	ITB	IUS	US
Even acknowledgment[1]	ACK 0	DLE (70)	DLE 0
Odd acknowledgment[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	
Transparent DLE	XDLE	DLE DLE	

[1] Change-of-direction character

Figure 9-3. Control Characters for BSC

BSC Character Synchronization (SYN SYN)

A BSC adapter without the internal clock receives timing pulses from the modem which, between itself and the transmitting adapter, establishes and maintains bit synchronization. The adapter that is starting to transmit, automatically sends two SYNs for establishing character synchronization at the receiving adapter. The receiving adapter establishes character synchronization by decoding the two consecutive SYNs. To maintain character synchronization, the transmitting adapter inserts a synchronization pattern, SYN SYN, for each transmit time-out. The synchronization pattern does not enter BCC or main storage. In transparent mode, the transparent synchronous idle (DLE SYN) is used.

An adapter with the internal clock establishes and maintains character synchronization on its own. For compatibility with this feature, the BSC adapter automatically sends two additional hexadecimal 55s preceding the character synchronization pattern.

Framing the BSC Message

The program at the transmitting station must frame the data to be sent with the correct line-control characters. These characters are stored at the receiving station, so the program at the receiving station must permit space for them in storage. When transmitting, the BSC adapter automatically generates and transmits SYN, pad, and CRC characters (LRC/VRC for ASCII) as needed for establishing and maintaining synchronization with the remote station and for error checking. When receiving, the BSC adapter removes all SYN and CRC characters (LRC/VRC for ASCII) and some pad characters from the data being sent to storage. The pad character following an NAK or EOT *is not* removed by the adapter.

Response characters (ACK 0, ACK 1, WACK, and NAK) are inserted by the program at the transmitting BSC adapter and not by the transmitting BSC adapter itself, and they are not deleted by the receiving BSC adapter. The program at the receiving BSC adapter must store these characters in a known location so that this program can test them to determine what action to take next.

BSC OPERATIONS

All BSC operations on the communications line are controlled through a combination of instructions in the system processing unit and the automatic controls started by the line-control characters.

Enable/Disable BSC

Enable BSC adapter sets on the 'data terminal ready' line to the modem; disable BSC adapter sets off the 'data terminal ready' line and resets the BSC adapter. The 'power on reset', 'system reset', or 'IPL' line also sets off the 'data terminal ready' line and resets the BSC adapter.

Since the 'data terminal ready' line controls switching of the modem to the data link, enable BSC adapter is a prerequisite to making a switched network connection. Disable BSC adapter is used to disconnect from a switched network.

Initialization Sequences

Initialization sequences, transmitted by the transmit and receive instructions, are described in *General Information—Binary Synchronous Communications*, GA27-3004. The data link (point-to-point nonswitched, point-to-point switched, or multipoint) determines the type of receive initial operation; these operations are described in the following paragraphs.

Receive Initial Operation (Point-to-Point Nonswitched)

On a nonswitched network, receive initial causes the BSC adapter to search for character synchronization. When character synchronization is made, *receive time-out* becomes active and the received data (starting with the first non-SYN character) is stored in the main storage area specified by the data buffer address. The operation ends and an IOB is posted complete when a change-of-direction character is received, the receive buffer length is zero, or a receive time-out occurs.

Receive Initial Operation (Point-to-Point Switched)

On a switched network, a receive initial operation conditions the BSC adapter. When the 'data set ready' line goes active, receive time-out becomes effective and the BSC adapter attempts to establish synchronization.

When character synchronization is made, the received data (starting with the first non-SYN character) is stored in the main storage area specified by the data buffer address. The receive buffer length is decreased each time a character is received. The operation ends and an IOB is posted complete when a change-of-direction character is received, the receive buffer length is zero, or a receive time-out occurs. In the case of a receive time-out, the recovery procedure is to issue the receive only.

Receive Initial Operation (Multipoint)

Receive initial is used to receive polling and selection sequences on a multipoint network. The receive buffer length should be loaded with one less than the maximum number of characters in the polling/selection sequence. A 2-character station address is used. For this operation, the address character must be loaded in the station address field of the IOB. The EBCDIC 2-bit or the ASCII 6-bit of the first station address character received is ignored; however, both characters of the address must be the same.

For example, assuming EBCDIC, if the station address field is loaded with either B or S, the adapter recognizes either BB or SS as the station address.

The basic mode of BSC is monitor mode for this operation. In this mode, the BSC adapter searches for character synchronization. When character synchronization is completed, 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 BSC adapter monitors for its station address. If it is not found, the BSC adapter continues monitoring the line. A decoded SOH or STX drops control mode and puts the BSC adapter back into monitor mode. If the station address is decoded as the first non-SYN characters after establishing character synchronization in control mode, the BSC adapter immediately enters addressed mode and puts the sequence, starting with the second station address character, into the main storage area specified by the data buffer address. The operation ends and the IOB is posted complete when a change-of-direction character is received, the receive buffer length is zero, or a receive time-out occurs.

Receive Initial Delayed Operation (Multipoint)

The receive initial delayed operation is the same as a receive initial operation except for the following:

- When the receive initial delayed command is issued and decoded, and the parameters for the command are put into control storage, the command is not executed until an end of transmission (EOT) character is received.
- When the EOT character is received, the parameters for the receive initial delayed command are taken from control storage, and the command is executed as a receive initial command. However, the receive initial operation starts in control mode where an EOT character does not have to be received.

The reason for the receive initial delayed operation is the same as for the receive initial operation; it is the quick response time needed between when the EOT character is received to when the next polling/selection sequence starts. For more information on the receive initial delayed operation, see *Receive Initial Operation (Multipoint)*.

Transmit and Receive Operation

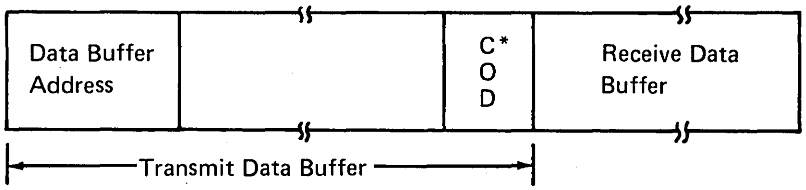
If a response results from the transmit operation, the combined transmit and receive operation must be used.

The transmit and receive operation is used for any type of transmission; that is, for control sequences or text data. It sets the BSC adapter to transmit mode, then takes characters from main storage and transmits them on the line. BCC accumulation, data mode, and transparent mode are set if the correct line-control characters are taken from storage. The transmit buffer length is decreased each time a character is transmitted and transmission continues until the transmit buffer length is zero. When the transmit buffer length is zero, the adapter is turned around to receive mode under the same operation.

In receive mode, the BSC adapter searches for character synchronization, then stores the characters received into main storage. As in transmit, the control characters received determine the function of the receive operation.

The operation ends and the IOB is posted complete when a change-of-direction sequence is received, the receive buffer length is zero, or a receive time-out occurs. At this time the completion code and status bytes in the IOB can be analyzed.

The reason for this combined transmit and receive instruction is the quick response time needed between the two operations. The effect of the data buffer address, the transmit buffer length, and the receive buffer length on the control sequences or text data is shown in Figure 9-4.



* COD = Change-of-direction character

The transmit and receive buffers must be allocated so that the receive buffer area follows the transmit buffer area, and is next to the transmit buffer area. The maximum size of the combined transmit and receive buffers is 4K bytes.

Figure 9-4. Main Storage Data Buffer at Start of Transmit and Receive Operation

The transmit and receive operation is used by both the primary and secondary station; that is, to send data and receive the response, and to send the response and receive data.

At the start of the transmit and receive operation, the adapter sends hexadecimal 55 (two additional hexadecimal 55s if the Internal Clock feature is installed), and two SYN characters. During transmit, the BSC adapter inserts the synchronization pattern, SYN SYN, for each transmit time-out. SYN is not part of the BCC and does not enter main storage. BCC compare takes place when an ITB, ETB, or ETX is received.

If the adapter enters data mode by receiving an STX or SOH, then only ETB, ETX, and ENQ are valid change-of-direction sequences. Outside of data mode, all turnaround sequences are valid change-of-direction sequences and will terminate the operation. The IOB is posted complete before the receive buffer length is equal to zero if a change-of-direction sequence is received.

Transmit and Receive Initial Operation (Multipoint)

The transmit and receive initial operation is the same as a transmit and receive operation except for the following:

- After the transmit buffer reaches 0, it is necessary to transmit an end of transmission (EOT) character.

- Bit 7 of the command modifier must be set on to enter receive initial mode. When bit 7 is on, the microcode looks for a polling sequence or an address without a preceding EOT character. In other words, the EOT character does not have to precede the polling/selection sequence from the primary station.

The reason for the transmit and receive initial operation is the same as for the transmit and receive operation; it is the quick response time needed between the two operations. For more information on the transmit and receive initial operation, see *Transmit and Receive Operation.*

ITB Operation

The IUS (intermediate unit separator) character is interpreted as the ITB control character to activate the ITB function. The primary station sends the BCC after the ITB and the secondary station receives and compares it; both stations then transmit more data with no line turnaround.

For nontransparent data, the primary station can transmit all ITB blocks in a single transmit and receive instruction.

When the secondary station receives an ITB character, the adapter remains in receive mode and receives the next ITB block. This continues until a change-of-direction character is recognized. When the ending sequence (ETB, ETX, or ENQ) is received, it is stored and the IOB is posted complete. At this time, the program checks the completion code and status bytes to determine the correct response.

Transparent Operation

In transmitting and receiving data, transparent mode is set by the DLE STX sequence. In transparent mode, the transmitting adapter automatically inserts a second DLE preceding each DLE from storage (except DLE STX), which is deleted by the receiving BSC adapter. The additional DLE does not enter BCC accumulation.

Either ETB, ETX, or ENQ (all change-of-direction characters) ends transparent mode at the primary station if it is at a location one less than the start of the receive buffer length. Therefore, the primary station inserts a DLE so that the single DLE followed by ETB, ETX, or ENQ informs the secondary station to leave transparent mode. This DLE is deleted by the secondary station and is not included in the BCC at either station.

Disconnect Operation

The program performs a disconnect operation on a switched network by giving a disable BSC adapter which drops the 'data terminal ready' line to the modem. The transmitting station sends a DLE EOT sequence with a transmit operation to inform the receiving station that it is going *on-hook.* A received DLE EOT sequence at the secondary station causes a disconnect operation.

Receive Operation

The receive operation is used when it is necessary to perform a receive operation after the end of the preceding instruction, such as when a receive time-out has occurred. The operation is the same as the receive part of the transmit and receive operation.

This instruction must be used after a receive time-out during a receive initial operation on a switched network or after a receive time-out during a transmit and receive.

2-Second Time-out

This control code function supplies a 2-second delay before transmitting a TTD or WACK. The start 2-second time-out must be given only with the control instruction. When the time-out is completed, the IOB is posted complete. The BSC adapter does not need to be enabled to perform the 2-second time-out operation.

The 2-second time-out is forced to end if a transmit command is issued while the 2-second time-out is executing.

DATA CHECKING AND BSC STATUS BYTES

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 BSC adapter compares these block check characters that it receives from the line with the block check characters that it generated. If the block check characters generated by the local communications adapter do not match the block check characters received from the line, the block check status bit is set (bit 1 of Figure 9-5). While servicing a completed IOB, the program must sample the status bits and determine if a block check has occurred.

If the IOB completion 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 acknowledgment characters alternate; ACK 0 is transmitted in response to even-numbered blocks and ACK 1 is transmitted in response to odd-numbered blocks. The program must transmit the correct positive acknowledgment. The first block of text transmitted is always an odd-numbered block. If the wrong acknowledgment character is returned, the primary station assumes that a block of data was lost and an error recovery procedure is started.

When block checking is started by ITB, the result of the block check compare is not transmitted immediately. Instead, if the block check compare is equal, the BSC adapter continues to receive and store characters. If the block check compare is not equal, the data check status bit is set on to indicate that a block check compare error occurred. When the next ETB or ETX character is received, it is stored and the IOB is posted complete. The status bits are tested to determine if all data was received correctly. An ENQ character also terminates the receive operation.

Bit	Desription When Bit Is Set
0	Time-out status: A receive time-out (3.25 seconds) occurred during a receive operation.
1	Block check during a receive operation. ● A CRC compare check occurred (EBCDIC). ● An LRC/VRC compare 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 BSC adapter.
2	Transmit adapter check: The BSC adapter did not move a character from main storage to the adapter before the next character had to be moved to accommodate the line. An overrun does not terminate the operation.
3	Receive adapter check: The BSC adapter did not move a character from the adapter to main storage before the next character had to be moved to accommodate the line. An overrun does not terminate the operation.
4	Invalid ASCII: BSC adapter found leftmost bit in an ASCII byte on during transmit operation.
5	Abortive disconnect: Indicates that the BSC adapter on a switched network was enabled, then the modem became ready, then not ready. This indicates the connection has been released and causes data terminal ready to turn off. The program must allow enough time for a forced disconnect to occur. The program can use the 2-second time-out to ensure this.
6	Not data set ready: Indicates that the modem is not ready to operate and that the BSC adapter is not enabled.
7	Not used.

Figure 9-5. BSC Adapter Status Byte

SUGGESTED ERROR RECOVERY PROCEDURES

If the error bit (bit 7) is on in the IOB completion code at the end of a transmit or a receive operation, the program should test the IOB status byte. Test the status bits and perform the procedures for recovering from the error in the order given in Figure 9-6. The program must check for lost data and analyze the last two characters received to find a response error.

If the data end address (IOB bytes 2 and 3) is more than the data buffer address and the receive data buffer length, a lost-data error is indicated.

Priority	Status Byte 0 Bit	Error Condition	Error Recovery Procedure (recommended program action)
1	6	Not data set ready	All cases—Action 1
2	4	Invalid ASCII character	All cases—Action 1
3	5	Abortive disconnect	All cases—Action 1
4	2 and 3	Adapter checks (transmit and receive)	Control mode—Action 5 Secondary—Action 4 Primary—Action 3
5	0	Receive time-out	Receive initial (switched)—Action 8 Control mode—Action 5 Secondary—Action 4 Primary—Action 3
6	1	CRC/LRC/VRC	Control mode—Action 5 Secondary—Action 2 Primary—Action 3
6	Program detected error[1]	Lost data	
7	Program detected error[1]	Abnormal response	Secondary: Absence of initial STX or terminal ETB/ETX—Action 4 Primary: Improper ACK immediately preceded by time-out—Action 6 Primary: Any response other than proper ACK or EOT—Action 7

[1] The program should provide lost-data detection.

Action Table

1. Permanent error occurred—operator must restart.
2. NAK was transmitted and received—retransmit data.
3. ENQ was transmitted and received—retransmit last response N times.
4. Issue receive portion of previous operation N times.
5. 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 primary 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 receive operation up to 6 times, then take Action 1.

The value of M should be equal to or greater than N.

The value of N should be a minimum of 7.

When M or N is reached, the error is a permanent error. On permanent errors, the program should cancel the job and tell the operator the nature of the error condition by means of an error message. Operator intervention is then required and the procedure is either to completely restart the job or to continue with the next job.

Note: A processor check stop causes an immediate cancel.

Figure 9-6. BSC Adapter Error Conditions and Recovery Procedures

BSC ERROR RECORDING

Parts of three sectors are reserved on the disk for recording BSC errors in either the BSC error history table or the BSC error counter table (also contains counts of I/O activity). The error history table (Figure 9-7) contains a 14-byte entry for each of the last 25 temporary or permanent BSC errors.

The error counter table (Figure 9-8) is a 92-byte entry containing the latest job totals and the cumulative totals for 14 different items. All counts in the error counter table are put into the table by SSP routines at end-of-job time.

Displacement of Leftmost Byte in Hex	Length in Bytes	Description
0	1	Command code
1	1	Command modifier
2	1	Sense information byte 0
3	1	Error retry count
4	1	Binary synchronous communication completion code
5	2	Terminal address
7	3	Date (yymmdd) on which the error occurred
A	4	Time of day (measured in timer units)

Figure 9-7. BSC Error History Table

Displacement of Leftmost Byte in Hex	Length in Bytes	Description
0	2	Number of job text blocks transmitted
2	4	Number of cumulative text blocks transmitted
6	2	Number of job text blocks received
8	4	Number of cumulative text blocks received
C	3	Date (yymmdd) on which the I/O counters in this table were reset through ERAP
F	1	Reserved
10	2	Number of job negative acknowledgements received
12	4	Number of cumulative negative acknowledgements received
16	2	Number of job data checks
18	4	Number of cumulative data checks
1C	2	Number of job forward aborts received
1E	4	Number of cumulative forward aborts received
22	2	Number of job aborts received
24	4	Number of cumulative aborts received
28	2	Number of job adapter checks during transmission
2A	4	Number of cumulative adapter checks during transmission
2E	2	Number of job adapter checks while receiving
30	4	Number of cumulative adapter checks while receiving
34	2	Number of job invalid responses received
36	4	Number of cumulative invalid responses received
3A	2	Number of job inquiries received as affirmative acknowledgements
3C	4	Number of cumulative inquiries received as affirmative acknowledgements

Figure 9-8 (Part 1 of 2). BSC Error Counter Table

Displacement of Leftmost Byte in Hex	Length in Bytes	Description
40	2	Number of job lost data errors
42	4	Number of cumulative lost data errors
46	2	Number of job disconnect time-outs
48	4	Number of cumulative disconnect time-outs
4C	2	Number of job receive time-outs
4E	4	Number of cumulative receive time-outs
52	2	Number of job transmission time-outs
54	4	Number of cumulative transmission time-outs
58	3	Date (yymmdd) on which the error counters in this table were reset through ERAP
5B	1	Reserved

Figure 9-8 (Part 2 of 2). BSC Error Counter Table

SDLC ADAPTER

Data that is transmitted or received by the SDLC (synchronous data link control) adapter is read from or written into main storage without any code translation. No code (such as EBCDIC or ASCII) is used; SDLC is bit oriented.

In addition, no control characters (such as ACK, NAK, and WACK used for BSC) are used to control the data link. The data link is controlled by the control field, which is part of the SDLC frame.

SDLC Frame

The SDLC frame transmits every command, every response, and all information over a data link using SDLC procedures. Each frame has a fixed format containing a starting flag (F), a station address field (A), a control field (C), an information field (I), which is optional, a frame check field (FC), and an ending flag (F). Therefore, those frames that contain an information field have a format of F, A, C, I, FC, F.

Figure 9-9 and the following paragraphs describe each field in the SDLC frame.

Flag (F, –, –, –, –, F)

There are two flags, starting and ending, for every SDLC frame. Both flags have a binary configuration of 01111110.

The starting flag, in addition to starting the frame, starts the transmission error checking. The ending flag ends the frame and the checking of transmission errors. When more than one frame is transmitted, the ending flag of one frame may also be the starting flag of the next frame.

For a frame to be valid, the number of bits in a frame between a starting and ending flag must be equal to or more than 32 bits. These 32 bits include the address field (8 bits), the control field (8 bits), and the frame check field (16 bits). The information field is not always permitted (see Figure 9-11), or it may be missing.

Continuous Flags: Continuous flags are automatically transmitted after a transmit-only operation has been completed or after a valid addressed frame is received with the poll bit on.

Station Address (F, A, –, –, –, F)

The address field is an 8-bit field that follows the starting flag in the frame format. This field always identifies System/34; the primary station is never identified in the address field.

System/34, in addition to recognizing its own address, can recognize the broadcast address (all 1's). The address field must be recognized before a frame can be received. The station address is specified in the line definition byte of the IOB.

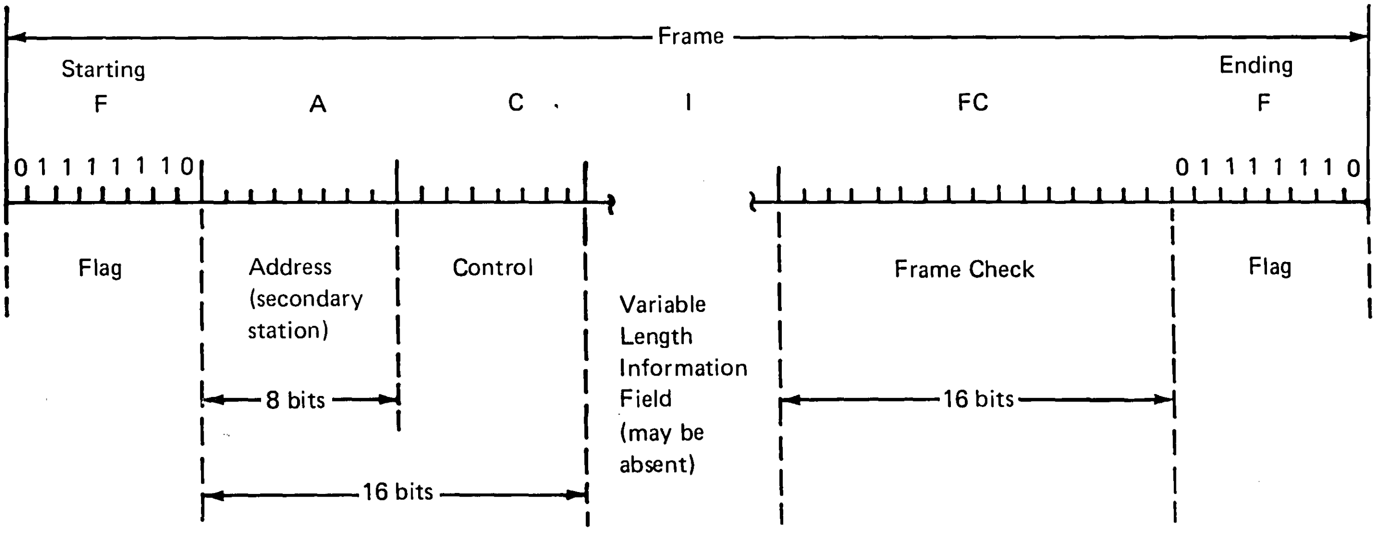


Figure 9-9. SDLC Transmission Frame

Control Field (F, A, C, –, –, F)

The control field is an 8-bit field that follows the station address field in the frame format. Both System/34 and the primary station use the control field to transfer information. System/34 also uses the control field for supervisory and nonsequenced responses; the primary station uses it for supervisory and nonsequenced commands.

The control field (see Figure 9-10) contains:

- Information for encoding the commands (from the primary station) and the responses (from System/34) needed to control the data link. (See *SDLC Commands and Responses* later in this chapter for a description of the commands and responses used by System/34.)
- A format identifier (bit 7 or bits 6 and 7) indicating if the frame is of the information transfer, supervisory, or nonsequenced format.
- A P/F (poll/final) bit. A poll bit is sent by the primary station to permit the transmission of data from System/34. System/34 sends a final bit in response to the poll bit when it has completed transmitting data. The P/F bit is always bit 3 of the control field.
- Either the sequence number of the frames that have been sent (Ns) or the sequence number of the next expected frame (Nr), or both.

Counting Sequenced Frames: When a station sends a sequenced frame (a frame with an information transfer format), the frame is counted in bits 4-6 of the control field. Similarly, when an error-free sequenced frame is received, the frame is counted in bits 0-2 of the control field. (Note that frames with a supervisory format contain the count of the frames received. This count is kept to ensure that frames are in sequence.)

The Nr count is always the count of the next expected frame; the next incoming Ns count is equal to the Nr count. If the incoming Ns count compares with the Nr count, the frame is in sequence and the Nr count advances. If the counts do not compare, the frame is out of sequence and the Nr count does not advance.

Up to seven frames may be sent before the receiving station must report its Nr count to the transmitting station. All transmitted frames must be kept by the transmitting station because a sequencing or line error may make it necessary to send them again.

Information Field (F, A, C, I, –, F)

This field, which follows the control field in the frame format, is not always included in the frame. Normally, a frame with an information transfer format contains an information field.

The information field can have any format or content; that is, it can include any bit sequence. However, the length of the field must be an integer number of 8-bit bytes not to exceed the buffering limits of the stations.

	0 1 2	3	4 5	6	7
Information Transfer Format	Nr[1]	P/F	Ns[2]		0
Supervisory Format	Nr	P/F	[3]	0	1
Nonsequenced Format	[4]	P/F	[4]	1	1

Poll/Final Bit

[1] Nr is the sequence number of the next expected frame.
[2] Ns is the sequence number of the last frame that was sent.
[3] Codes for supervisory commands/responses.
[4] Codes for nonsequenced commands/responses.

Figure 9-10. SDLC Control Field Format

Frame Check Field (F, A, C, I, FC, F)

The frame check field, which precedes the ending flag of the frame, contains 16 bits for the purpose of checking transmission accuracy. It supplies a cyclic redundancy check (CRC) to all bits in the frame except for the flags.

SDLC Commands and Responses

The commands and responses are specified in the bit configuration of the control field. When System/34 receives one of these bit configurations from the primary station, it is a command; when System/34 transmits to the primary station, it is a response.

The SDLC commands and responses are given in Figure 9-11; they are described with each of the three control field formats in the following paragraphs.

Information Transfer Format

This format is identified with a 0 in bit 7 of the control field. Frames with this format are used to transfer information over a data link.

Only those frames containing this format are sequenced; therefore, the control field must contain both the Nr and the Ns count fields (see Figure 9-10). These two count fields ensure that sequenced frames are not lost or duplicated. When a sequenced frame is transmitted, the transmitting station increases its Ns count by 1. The station receiving a valid, sequenced frame increases its Nr count by 1. For more information on the Nr and Ns counts, see *Counting Sequenced Frames* earlier in this chapter.

Format (See Note.)	Control Field Bit Configuration			Acronym	Command	Response	I-Field Not Permitted	Command/Response Description
	0 1 2	3	4 5 6 7					
I	Nr	P/F	Ns 0	I	X	X		Sequenced information frame
S	Nr	P/F	0 0 0 1	RR	X	X	X	Ready to receive
	Nr	P/F	0 1 0 1	RNR	X	X	X	Not ready to receive
NS	0 1 0	P	0 0 1 1	DISC	X		X	System/34 cannot receive or transmit information frames.
	0 1 1	F	0 0 1 1	NSA		X	X	System/34 acknowledges DISC or SNRM.
	1 0 0	P	0 0 1 1	SNRM	X		X	System/34 can transmit on command.
	1 1 1	P/F	0 0 1 1	TEST	X	X		Tests the transmission of data.
	1 0 0	F	0 1 1 1	CMDR		X		A nonvalid command was received by System/34; must receive a DISC or SNRM.
	1 0 1	P/F	1 1 1 1	XID	X	X		Exchange station identification.
	0 0 0	F	1 1 1 1	DM		X	X	System/34 is offline.

Note: I = Information, S = Supervisory, and NS = Nonsequenced

Figure 9-11. SDLC Commands and Responses

Supervisory Format

Bits 6 and 7 of the control field identify this format; they contain a 0 and a 1, respectively. The format is used to acknowledge information frames or to report a busy condition.

Bits 4 and 5 of the control field are used to encode the commands and the responses. The supervisory commands and responses are:

- RR (receive ready): Used as a command or a response. The transmitting station acknowledges the sequenced frames through the Nr count minus 1. This command/response also indicates that the transmitting station is ready to receive.
- RNR (receive not ready): Used as a command or a response. The transmitting station sends RNR to indicate a temporarily busy condition in which no frames that need buffer space can be received. Sequenced frames through Nr minus 1 are acknowledged.

Nonsequenced Format

This format is identified with 1's in bits 6 and 7 of the control field. It is used to perform data link control functions. Communications using the nonsequenced format are not sequence-checked; they do not use the Nr and Ns count field.

Excluding bit 3 (P/F) and bits 6 and 7 (format identifier), the other five bits are used for encoding the nonsequenced commands and responses. There are some nonsequenced commands that need specific nonsequenced responses from System/34. These commands are SNRM (set normal response mode), DISC (disconnect), TEST, and XID (exchange station identification). A response from System/34 to one of these commands will occur before any other supervisory or information transfer format response.

If more than one nonsequenced command is received by System/34 before a response, the additional commands (more than one) are ignored. The response is to the first command received.

The commands and responses in the nonsequenced format for System/34 are:

- DISC (disconnect): This command places the receiving System/34 in a disconnected state. System/34 should respond with an NSA (nonsequenced acknowledgment); it should also disable the adapter when a DISC command is received. No information field is permitted with the disconnect command.
- NSA (nonsequenced acknowledgment): This is an affirmative response to an SNRM or DISC command; it acknowledges that the command was received. No information field is permitted with the NSA response.
- SNRM (set normal response mode): This command places System/34 in a normal response mode (NRM). System/34 remains in NRM until it receives a DISC command. The expected response to an SNRM command is NSA. A System/34 in NRM cannot transmit until it receives a frame with the poll bit on.
- TEST: This is a command from the primary station or a response from System/34. The primary station starts one round-trip transmission of test data to which System/34 responds; that is, the data that is sent to System/34 with a TEST command is normally returned with a TEST response from System/34 (unless the data was too long for the buffer, in which case the data is not returned). This command/response can contain an information field. The information field of the TEST response must be the same as the information field of the TEST command.
- CMDR (command reject): This is a response to an invalid command received by System/34. System/34 repeats the CMDR response until an SNRM or DISC command is received.

  A command is invalid if:

  - The command is not used at the receiving station.
  - The information field is too long for the buffer space that was permitted (except for the TEST command).
  - The Nr count is out of range.
  - An information field was sent with a command that does not permit an information field.

A CMDR response includes an information field that gives the reason for the rejected command. The format of this field includes:

First byte—A duplicate of the control field of the command that caused the CMDR response.

Second byte—The receiving station's Nr and Ns count fields as they were before sensing the reason for the CMDR.

Third byte—(0000wxyz)

0000 = Pad characters.

w = The Nr sequence count in byte 1 is out of range.

x = The information field was too long. (This bit is mutually exclusive with bit z.)

y = Received an information field that was not permitted. (Bit z must be on with this bit.)

z = An invalid command was received.

- XID (exchange station identification): This command/response is used by the primary station as a command to request station identification from the addressed System/34. The primary station can also give its own identification to the addressed System/34. System/34 uses this command/response only in response to a received XID command.
- DM (disconnected mode): This response is transmitted to the primary station to indicate that System/34 is in a disconnected state (normal disconnect mode), and System/34 requests an online status. No information field is permitted with this response.

SDLC Response Modes

There are two response modes for a System/34 using SDLC procedures–normal response mode (NRM) and normal disconnect mode (NDM). In NRM, System/34 can transmit if it has received a frame with the poll bit on; more than 1 frame can be transmitted. The last frame transmitted can have the final bit on, or a supervisory frame (RR or RNR), with the final bit, can follow the last information frame. Once a frame is transmitted with the final bit on, System/34 cannot transmit again until it receives another frame with the poll bit on.

In NDM, System/34 normally responds with DM (disconnected mode) unless it receives an SNRM, DISC, TEST, or XID command.

SDLC Transmission States

There are four transmission states for an SDLC data link–active, disconnect, idle, and transient. The data link can be in only one state at any one time.

Active State

When the data link is in the active state, a station is transmitting or receiving data. Flags are used to activate or maintain the active state. Once System/34 is in the active state, it must remain active until it sends a frame with the final bit on or until it must abort a frame.

Disconnect State

In the disconnect state when the data link is not operational; no transmissions are possible. The primary station does not monitor the data link for incoming transmissions.

Idle State

In the idle state, the data link is operational but there are no data transmissions. When a station does not have the priority to transmit, that station goes to the idle state.

Also, when 15 or more consecutive 1 bits are sensed, the data link goes to the idle state.

Transient State

When the data link is in the transient state, a station is getting ready to transmit; this is known as *turnaround delay*. The delay starts when a station sets the request to send signal on, and ends when the modem supplies the clear to send signal.

SDLC Input/Output Block

Each SDLC adapter operation is specified by an input/output block (IOB) located in main storage. The IOB, issued by the supervisor call input/output request instruction, contains all the information needed to perform a requested operation.

If more than one operation is to be performed, the IOBs must be queued by issuing a supervisory call input/output request instruction for each operation. At the end of each operation, the IOB is posted complete and the next operation on the IOB queue is started. See Figure 9-12 for a description of the SDLC IOB.

Posting IOBs Complete

A completion code is generated by the SDLC adapter when it becomes necessary to inform the system program of empty or full transmit/receive buffers. Completion codes are also generated when an operation ends or as a result of an error condition during a transmit or receive operation.

On a receive operation, an IOB is posted complete if:

- A single valid addressed frame is sensed.
- An addressed invalid frame is sensed.
- An addressed valid frame is sensed but with wrong frame checking.
- The inactivity timer has timed out.
- An adapter check has occurred.

On a transmit operation, an IOB is posted complete if a frame has been sent with a transmit only or a transmit final instruction.

An IOB is posted complete for either a transmit or a receive operation if:

- An abortive disconnect occurs on a switched line.
- A buffer overrun occurs.
- The transmit data buffer length is 0 on a transmit operation, or the receive data buffer length is 0 on a receive operation and the trailing flag or abort condition is recognized. Once the receive data buffer length is 0, there is no more data transferred to storage but the adapter continues to collect frame check characters on the incoming data while monitoring for a trailing flag or an abort condition. (The receive and transmit data buffer lengths are decreased by the SDLC adapter as data is received or transmitted.)
- An adapter check has occurred.

Displacement of Leftmost Byte in Hex	IBM Program Label	Length in Bytes	Description
0	SIOBECM	1	Event control mask *Hex* *Meaning* 80 Do not skip indicator. 40 Data buffer address is real.
1	SIOBHCMP	1	Completion code *Hex* *Meaning* 80 Active: Set on by control storage when processing the IOB, and set off by control storage when processing is complete. If this bit is on, processing of the IOB is not permitted. 40 Complete: Set on by control storage when the IOB is complete; set off by main storage. If this bit is on, processing of the IOB is not permitted. 10 Hold: Set on by control storage when processing the IOB. Set on by main storage to inhibit processing of the IOB; the supervisor call test/set command is used to set this bit from main storage. 01 Error detected: Set on if any bit in status byte 0 is on.
2	SIOBPARM	1	Parameter byte
3	SIOBQ	1	Command (Q) code Bits 0 1 2 3 = Attachment address Bits 4 5 6 7 = Command type 0 0 0 0 Control 0 0 0 1 Receive 0 0 1 0 Transmit/receive: When the transmit operation ends; processing of the IOBs will continue from the beginning of the queue; a receive IOB must be at the beginning of the queue. 0 0 1 1 Receive initial 0 1 0 0 Transmit final 0 1 0 1 Transmit only 0 1 1 0 Reserved 0 1 1 1 Receive delayed *Note:* The SDLC control storage program does not check the validity of the command code. .The program decodes the command from only the low-order 3 bits and proceeds with the operation. Where the low-order 3 bits are defined to be reserved, the operation defaults to a receive operation.

Figure 9-12 (Part 1 of 3). SDLC IOB

Displacement of Leftmost Byte in Hex	IBM Program Label	Length in Bytes	Description
4	SIOBR	1	Command modifier When the command (Q) code is hex 0: C0 = Control enable 80 = Control disable When the command (Q) code is not hex 0, the command modifier bits are reserved.
5	SIOBSTA@	1	Station address: Address field in the SDLC frame that identifies System/34.
6	SIOBBUF@	2	Data buffer address: Points to the beginning of the data buffer.
8	SIOBBUFL	2	Data buffer length: Defines the number of bytes in the data buffer.
A	SIOBST0		Status byte 0 *Hex* *Meaning* 80 Time-out 40 Frame check 20 Adapter check 10 Buffer overrun (receive) 08 Invalid frame 04 Abortive disconnect 02 Not data set ready 01 Idle time-out (primary station)
B	SIOBST1	1	Status byte 1 (reserved)
C	SIOBDEA	2	Data end address: Indicates the last position of the buffer (plus 1 position) that was used at the completion of a transmit or receive command.
E	SIOBSTAT	2	Reserved
10	SIOBTCB	2	Task control block address
12	SIOBQUE	1	Queue identification

Figure 9-12 (Part 2 of 3). SDLC IOB

Displacement of Leftmost Byte in Hex	IBM Program Label	Length in Bytes	Description
13	SIOBLDEF	1	Line definition byte *Hex Meaning* 80 Half rate is selected. 40 Internal clock is needed. 20 IBM modem is installed. 10 Answer tone must be supplied by data terminal equipment. 08 Standby line is selected. 04 Multipoint line is selected. 02 Switched data terminal ready line is selected. 01 Nonswitched line is selected.
14	SIOBRES	4	Reserved

Figure 9-12 (Part 3 of 3). SDLC IOB

Main Storage Data Areas

The transmit buffer and the receive buffer are main storage data areas used by SDLC when data is transmitted or received over the data link.

Transmit Buffer

The transmit buffer, shown in Figure 9-13, contains the control field and information field for one frame to be transmitted by the SDLC adapter. During the transmit operation, the adapter reads and transmits one byte at a time from the transmit buffer.

The data buffer address (DBA) and the transmit data buffer length must be specified in the SDLC IOB, and the data to be transmitted must be stored in the buffer before the transmit operation is issued.

DBA	Transmit Data Buffer Length
Control	Information

Figure 9-13. Transmit Buffer at Start of Transmit Operation

Receive Buffer

The receive buffer, shown in Figure 9-14, contains the control field and information field received in one frame. During the receive operation, the SDLC adapter fills the receive buffer one byte at a time with data received on the data link.

The data buffer address (DBA) and the receive data buffer length must be specified in the SDLC IOB before the receive operation is issued.

DBA	Receive Data Buffer Length
Control	Information

Figure 9-14. Receive Buffer at Start of Receive Operation

SDLC OPERATIONS

All operations on the data link are controlled by input/output blocks issued by supervisor call input/output request instructions.

Enable/Disable SDLC

The enable SDLC operation enables the SDLC adapter and sets on the 'data terminal ready' line to the modem if a nonswitched network configuration or a switched DTR (data terminal ready) configuration is specified in the line definition byte of the IOB.

The disable SDLC operation disables the SDLC adapter and sets off the 'data terminal ready' line to the modem. The 'power on reset', 'system reset', or 'IPL' line also disables the SDLC adapter and sets off the 'data terminal ready' line.

Receive Initial Operation

The receive initial operation causes an SDLC initialization sequence to be performed and the first frame received to be loaded into the receive buffer. The data line selected (switched DTR or nonswitched) determines the type of initialization sequence.

Receive Initial Operation (Switched DTR Network)

On a switched DTR network, the receive initial operation causes the SDLC adapter to wait for the 'data set ready' line to be set on. When the 'data set ready' line is set on, the SDLC adapter (1) starts the inactivity counter, (2) generates an answer tone if one is needed, and (3) loads the receive buffer with the data received on the data link.

When the receive operation is completed, the IOB is posted complete. At this time the bits in the completion code byte can be checked.

Receive Initial Operation (Nonswitched Network)

On a nonswitched network, the receive initial operation causes the SDLC adapter to load the receive buffer with the data received on the data link. The adapter does not start the inactivity counter.

When the receive operation is completed, the IOB is posted complete. At this time the bits in the completion code byte can be checked.

Receive Operation

The receive operation causes the SDLC adapter to start the inactivity counter and to load the receive buffer with data received on the data link.

When a receive operation has been completed, the IOB is posted complete. At this time the status bits in the completion code byte can be checked.

Transmit Only Operation

The transmit only operation causes the SDLC adapter to transmit the data in the transmit buffer. When the transmit operation has been completed, the adapter holds the line in the active state by sending continuous flag bytes until another transmit operation starts. The transmit only instruction lets the adapter transmit continuous frames without any receive operations between the frames.

When the transmit operation has been completed, the IOB is posted complete. At this time the status bits in the completion code byte can be checked.

Transmit Operation (Poll/Final Bit On)

The transmit operation causes the SDLC adapter to transmit the data in the transmit buffer and prepare for line turnaround. The transmit operation is then followed by a receive operation.

When the transmit operation has been completed, the IOB is posted complete. At this time the completion code can be checked. An IOB for a receive command must be on the queue.

Transmit Final Operation

The transmit final operation causes the SDLC adapter to (1) transmit the data in the transmit buffer and (2) generate the completion code. This operation can be used when no response is needed to a final transmitted message. When the IOB is posted complete, the completion code can be checked.

Receive Delayed Operation

The receive delayed operation queues receive operations which are not to be executed immediately to the SDLC adapter. A transmit operation is needed before a receive delayed is executed. After a transmit operation is issued, executed, and posted complete, the adapter will execute the queued receive delayed. This operation ensures there is a receive available as soon as the transmit is complete.

When the receive operation is completed, the IOB is posted complete. At this time the bits in the completion code byte can be checked.

SDLC STATUS BYTES

The results of a transmit or a receive operation can be determined by checking the status bytes of the SDLC IOB. Figure 9-15 and the following paragraphs describe each bit of these status bytes.

Byte 0	Bit	Meaning	Bit Set On When:
	0	Time-out	The activity timer is completed.
	1	Frame check	A valid addressed frame is detected with an invalid frame check.
	2	Adapter check	A character was not moved to or from main storage before the next character had to be moved to accomodate the line.
	3	Receive buffer overrun	The receive buffer was not long enough to accomodate the incoming frame.
	4	Invalid frame	Any of the following occurs: ● A flag is detected off a byte boundary. ● An ending flag is detected within 32 bits of the starting flag. ● An abort sequence is detected. ● An idle condition is detected between a starting flag and an ending flag.
	5	Abortive disconnect	The data set ready line comes on and then goes off on a switched line.
	6	Data set not ready	An operation end interrupt is generated on a leased line if 'data set ready' is not on or does not come on 3 seconds after a supervisor call is issued.

Byte 1—Not used
Bytes 2 and 3—Data end address

Figure 9-15. SDLC Status Bytes

Inactivity Timer

The inactivity timer is used by the SDLC adapter to prevent long periods of inactivity that might result from an error condition.

Error conditions causing inactivity let the timer run out (time-out), and a completion code to be generated with the time-out status bit set (bit 0 of status byte 0).

The timer is started at the start of a receive operation. If a complete frame (valid or not valid) is not received in 32 seconds (time of the inactivity timer), the time-out condition is posted.

Adapter Checks

Adapter checks can occur on a receive operation or a transmit operation. On a receive operation, an adapter check occurs if another character is received before the preceding character is sensed. On a transmit operation, a check occurs if it is time to transmit a character but no character was loaded into the transmit buffer.

An adapter check on a receive operation is identified by bit 3 of status byte 0; bit 2 of status byte 0 identifies an adapter check on a transmit operation.

Invalid Frame: A frame is invalid if any of the following occur after a starting flag is sensed on a receive operation:

- An ending flag is sensed in less than 32 bits after a starting flag was sensed.
- A flag is sensed off a byte boundary.
- An abort sequence is sensed.
- An idle condition occurs.

An invalid frame is identified by bit 4 of status byte 0.

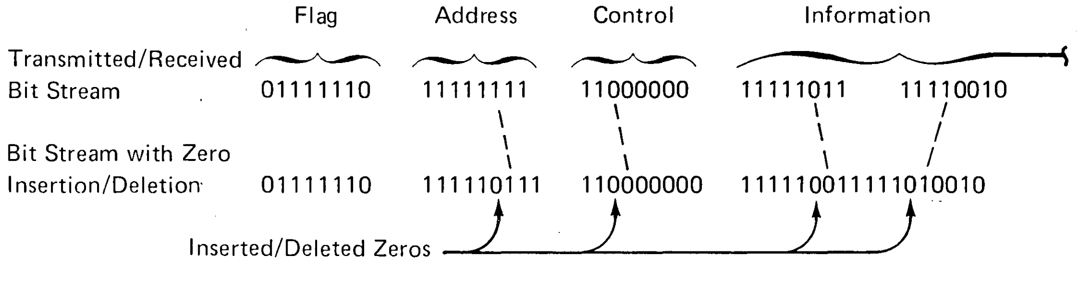
Abortive Disconnect: An abortive disconnect occurs when the 'data set ready' line goes not active on a switched line; bit 5 of status byte 0 is set on.

Data Set (Not) Ready: An operation end interrupt is generated on a leased line if 'data set ready' is not on or does not come on in three seconds after a supervisor call is issued.

ZERO BIT INSERTION/DELETION

Zero bit insertion/deletion ensures that bit streams that are the same as the flag are not transmitted in the address, control, information, and frame check fields of the frame; this is done in transmit mode by inserting a binary 0 bit into the data stream after five consecutive 1 bits (see Figure 9-16).

In receive mode, a 0 bit following five consecutive 1 bits is deleted. If the bit (call it bit 6) following five consecutive 1 bits is also a 1 bit, the bit stream is either a flag or an error. The next bit must be checked to determine if the data stream is a flag or an error. If the next bit (call it bit 7) is a 0 bit, the bit stream is received as a flag but a 1 bit indicates an error.



Note: No 0's are inserted in the flag field.

Figure 9-16. Zero Bit Insertion/Deletion

NRZI TRANSMISSION CODING

Because SDLC is bit oriented, it is important to maintain bit synchronization. This is the function of NRZI (zeros complemented transition coding).

NRZI prevents the extended periods of transitionless data when consecutive 0 bits are transmitted by changing the state of the data (from + to -, or from - to +) when transmitting a 0 bit. The data is not changed when 1 bits are transmitted (see Figure 9-17). As a result, continuous transitions occur for consecutive 0 bits and no transitions occur for consecutive 1 bits.

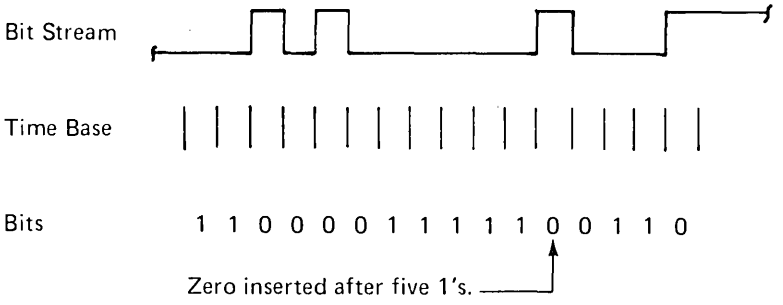


Figure 9-17. NRZI Transmission Coding

Zero bit insertion/deletion causes transitions by inserting a binary 0 into the data stream after five consecutive 1 bits. Therefore, a transition must occur after the transmission of no more than five 1 bits (except for a flag).

NRZI must be used with modems that are not synchronous (needing an internal clock) and with synchronous modems that are sensitive to transitionless bit streams. If a synchronous modem is being used and its sensitivity to transitionless bit streams cannot be determined, the user should contact his IBM marketing representative.

The internal clock must be used with modems (data sets) that do not supply clocking to the adapter. When the internal clock and NRZI transmission coding are both used, sixteen 0 bits are inserted into the data stream in front of the starting flag. Insertion of these 0 bits supplies 16 transitions that ensure initial bit synchronization.

Note: All DTEs (data terminal equipment) on the same data link must use the same encoding/decoding method (NRZI or non-NRZI). Failure to use the same method results in no communications between the DTEs.

SDLC ERROR RECORDING

Parts of three sectors are reserved on the disk for recording SDLC errors in either the SDLC error history table or the SDLC error counter table. (The SDLC error counter table also contains counts of I/O activity.) The error history table (Figure 9-18) contains a 14-byte entry for each of the last 25 temporary or permanent SDLC errors.

The error counter table (Figure 9-19) is a 80-byte entry containing the latest job totals and the cumulative totals for twelve different items. The latest job counts for all nine items are put into the table by SSP routines at end-of-job time; the cumulative counts for all nine items are updated by the SSP routines.

Displacement of Leftmost Byte in Hex	Length in Bytes	Description
0	1	Q-byte of the operation
1	1	Sense information byte 0
2	1	Sense information byte 1
3	1	SDLC control field
4	1	SDLC station address field
5	1	Queue header
6	3	Date (yymmdd) on which the error occurred
9	1	Reserved
A	4	Time of day (measured in timer units)

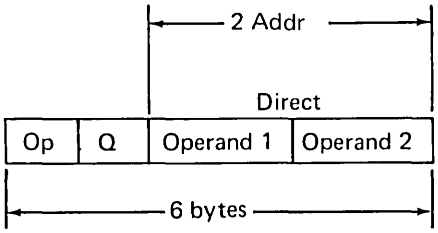
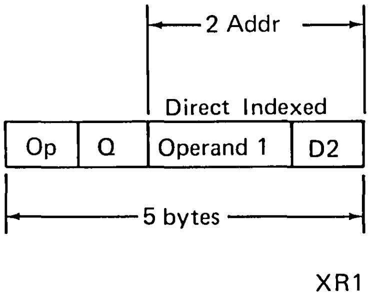
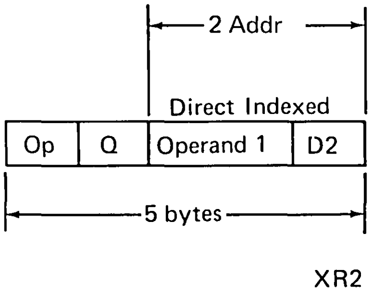
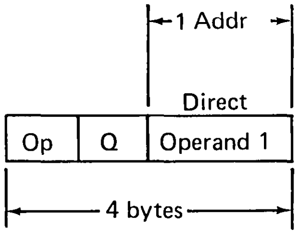
Figure 9-18. SDLC Error History Table

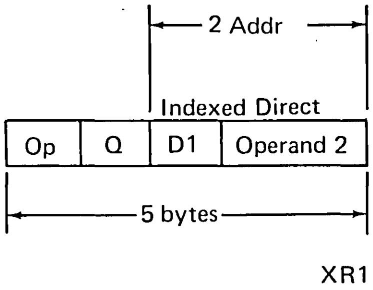
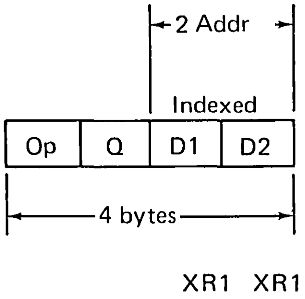
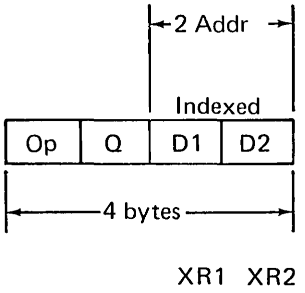
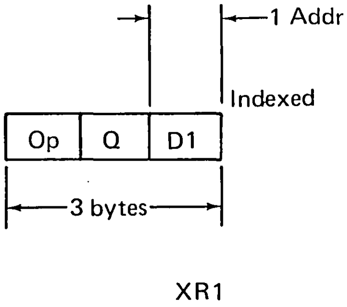
Displacement of Leftmost Byte in Hex	Length in Bytes	Description
0	2	Number of job information frames transmitted
2	4	Number of cumulative information frames transmitted
6	2	Number of job information frames retransmitted
8	4	Number of cumulative information frames retransmitted
C	2	Number of job information frames received
E	4	Number of cumulative information frames received
12	2	Number of job information frames purged
14	4	Number of cumulative information frames purged
18	2	Number of job total frames transmitted
1A	4	Number of cumulative total frames transmitted
1E	2	Number of job total frames received
20	4	Number of cumulative total frames received
24	3	Date (yymmdd) on which the I/O counters in this table were reset through ERAP
27	1	Reserved
28	2	Number of job block check errors
2A	4	Number of cumulative block check errors
2E	2	Number of job invalid frame errors
30	4	Number of cumulative invalid frame errors
34	2	Number of job abortive disconnect time-outs
36	4	Number of cumulative abortive disconnect time-outs
3A	2	Number of job receive time-outs
3C	4	Number of cumulative receive time-outs
40	2	Number of job adapter checks
42	4	Number of cumulative adapter checks

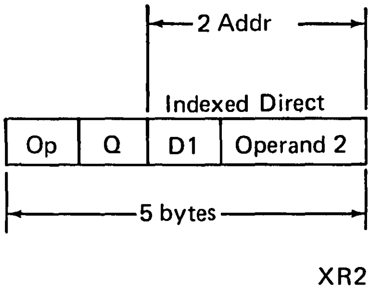
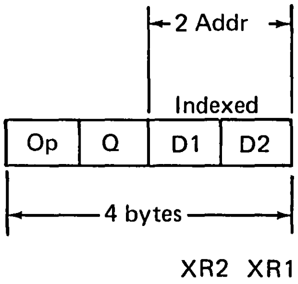
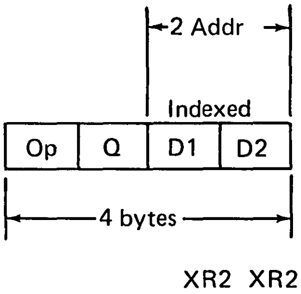
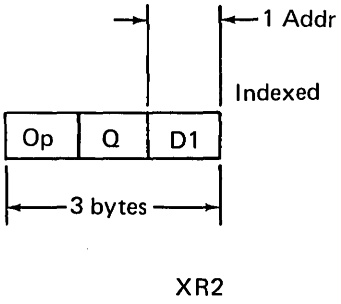
Figure 9-19 (Part 1 of 2). SDLC Error Counter Table

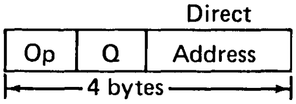
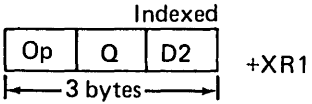
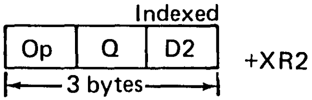
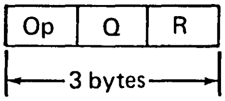
Displacement of Leftmost Byte in Hex	Length in Bytes	Description
46	2	Number of job idle detect time-outs
48	4	Number of cumulative idle detect time-outs
4C	3	Date (yymmdd) on which the error counters in this table were reset through ERAP
4F	1	Reserved

Figure 9-19 (Part 2 of 2). SDLC Error Counter Table

Op Code	Type
04 06 07 08 0A 0B 0C 0D 0E 0F	 2 Addr Direct Op Q Operand 1 Operand 2 6 bytes
14 16 17 18 1A 1B 1C 1D 1E 1F	 2 Addr Direct Indexed Op Q Operand 1 D2 5 bytes XR1
24 26 27 28 2A 2B 2C 2D 2E 2F	 2 Addr Direct Indexed Op Q Operand 1 D2 5 bytes XR2
34 35 36 38 39 3A 3B 3C 3D	

Op Code	Type
44 46 47 48 4A 4B 4C 4D 4E 4F	 2 Addr Indexed Direct Op Q D1 Operand 2 5 bytes XR1
54 56 57 58 5A 5B 5C 5D 5E 5F	 2 Addr Indexed Op Q D1 D2 4 bytes XR1 XR1
64 66 67 68 6A 6B 6C 6D 6E 6F	 2 Addr Indexed Op Q D1 D2 4 bytes XR1 XR2
74 75 76 78 79 7A 7B 7C 7D	

Op Code	Type
84 86 87 88 8A 8B 8C 8D 8E 8F	 2 Addr Indexed Direct Op Q D1 Operand 2 5 bytes XR2
94 96 97 98 9A 9B 9C 9D 9E 9F	 2 Addr Indexed Op Q D1 D2 4 bytes XR2 XR1
A4 A6 A7 A8 AA AB AC AD AE AF	 2 Addr Indexed Op Q D1 D2 4 bytes XR2 XR2
B4 B5 B6 B8 B9 BA BB BC BD	

Op Code	Type
C0 C2	
D0 D2	
E0 E2	
F0 F1 F2 F4 F5 F6	

Appendix B. EBCDIC Code Meanings

Below are definitions of the column headings used in the following table.

Hex Value. The internal EBCDIC code used by the system, expressed as a hexadecimal notation.

Binary Value. The internal EBCDIC code expressed as a binary notation.

Print Graphic. The graphic printed by this system for the EBCDIC code shown. For example, graphics printed for the EBCDIC code stored in the print data field correspond to the binary values shown in the chart. Hence, a main storage value of hex 6C is printed as %.

Display Screen Graphic. This column shows the graphic that is displayed on the display screen for the associated main storage EBCDIC code shown in the binary value column. A main storage value of hex 50 is displayed as &.

Hex Value	Binary Value	Print Graphic	Display Screen Graphic
00	00000000		
01	00000001		
02	00000010		
03	00000011		
04	00000100		
05	00000101		
06	00000110		
07	00000111		
08	00001000		
09	00001001		
0A	00001010		
0B	00001011		
0C	00001100		
0D	00001101		
0E	00001110		
0F	00001111		
10	00010000		
11	00010001		
12	00010010		
13	00010011		
14	00010100		
15	00010101		
16	00010110		
17	00010111		
18	00011000		
19	00011001		
1A	00011010		
1B	00011011		
1C	00011100		$\overline{*}$
1D	00011101		
1E	00011110		
1F	00011111		
20	00100000		
21	00100001		
22	00100010		
23	00100011		
24	00100100		
25	00100101		
26	00100110		
27	00100111		
28	00101000		
29	00101001		
2A	00101010		
2B	00101011		
2C	00101100		
2D	00101101		
2E	00101110		
2F	00101111		
30	00110000		
31	00110001		
32	00110010		
33	00110011		
34	00110100		
35	00110101		
36	00110110		
37	00110111		
38	00111000		
39	00111001		
3A	00111010		
3B	00111011		
3C	00111100		
3D	00111101		
3E	00111110		
3F	00111111		
40	01000000	Blank	Blank
41	01000001		
42	01000010		
43	01000011		
44	01000100		
45	01000101		
46	01000110		
47	01000111		
48	01001000		
49	01001001		
4A	01001010	¢	¢
4B	01001011	.	.
4C	01001100	<	<
4D	01001101	(	(
4E	01001110	+	+
4F	01001111	|	|
50	01010000	&	&
51	01010001		
52	01010010		
53	01010011		
54	01010100		
55	01010101		
56	01010110		
57	01010111		
58	01011000		
59	01011001		
5A	01011010	!	!
5B	01011011	$	$
5C	01011100	*	*
5D	01011101	)	)
5E	01011110	;	;
5F	01011111	¬	¬

Hex Value	Binary Value	Print Graphic	Display Screen Graphic
60	01100000	—	—
61	01100001	/	/
62	01100010		
63	01100011		
64	01100100		
65	01100101		
66	01100110		
67	01100111		
68	01101000		
69	01101001		
6A	01101010	¦	¦
6B	01101011	,	,
6C	01101100	%	%
6D	01101101	—	—
6E	01101110	>	>
6F	01101111	?	?
70	01110000		
71	01110001		
72	01110010		
73	01110011		
74	01110100		
75	01110101		
76	01110110		
77	01110111		
78	01111000		
79	01111001	‘	‘
7A	01111010	:	:
7B	01111011	#	#
7C	01111100	@	@
7D	01111101	'	'
7E	01111110	=	=
7F	01111111	"	"
80	10000000		
81	10000001	a	a
82	10000010	b	b
83	10000011	c	c
84	10000100	d	d
85	10000101	e	e
86	10000110	f	f
87	10000111	g	g
88	10001000	h	h
89	10001001	i	i
8A	10001010		
8B	10001011		
8C	10001100		
8D	10001101		
8E	10001110		
8F	10001111		

Hex Value	Binary Value	Print Graphic	Display Screen Graphic
90	10010000		
91	10010001	j	j
92	10010010	k	k
93	10010011	l	l
94	10010100	m	m
95	10010101	n	n
96	10010110	o	o
97	10010111	p	p
98	10011000	q	q
99	10011001	r	r
9A	10011010		
9B	10011011		
9C	10011100		
9D	10011101		
9E	10011110		
9F	10011111	■	
A0	10100000		
A1	10100001	~	~
A2	10100010	s	s
A3	10100011	t	t
A4	10100100	u	u
A5	10100101	v	v
A6	10100110	w	w
A7	10100111	x	x
A8	10101000	y	y
A9	10101001	z	z
AA	10101010		
AB	10101011		
AC	10101100		
AD	10101101		
AE	10101110		
AF	10101111		
B0	10110000		
B1	10110001		
B2	10110010		
B3	10110011		
B4	10110100		
B5	10110101		
B6	10110110		
B7	10110111		
B8	10111000		
B9	10111001		
BA	10111010		
BB	10111011		
BC	10111100		
BD	10111101		
BE	10111110		
BF	10111111		

Hex Value	Binary Value	Print Graphic	Display Screen Graphic
C0	11000000	{	{
C1	11000001	A	A
C2	11000010	B	B
C3	11000011	C	C
C4	11000100	D	D
C5	11000101	E	E
C6	11000110	F	F
C7	11000111	G	G
C8	11001000	H	H
C9	11001001	I	I
CA	11001010		
CB	11001011		
CC	11001100		
CD	11001101		
CE	11001110		
CF	11001111		
D0	11010000	}	}
D1	11010001	J	J
D2	11010010	K	K
D3	11010011	L	L
D4	11010100	M	M
D5	11010101	N	N
D6	11010110	O	O
D7	11010111	P	P
D8	11011000	Q	Q
D9	11011001	R	R
DA	11011010		
DB	11011011		
DC	11011100		
DD	11011101		
DE	11011110		
DF	11011111		
E0	11100000	\	\
E1	11100001		
E2	11100010	S	S
E3	11100011	T	T
E4	11100100	U	U
E5	11100101	V	V
E6	11100110	W	W
E7	11100111	X	X
E8	11101000	Y	Y
E9	11101001	Z	Z
EA	11101010		
EB	11101011		
EC	11101100		
ED	11101101		
EE	11101110		
EF	11101111		

Hex Value	Binary Value	Print Graphic	Display Screen Graphic
F0	11110000	0	0
F1	11110001	1	1
F2	11110010	2	2
F3	11110011	3	3
F4	11110100	4	4
F5	11110101	5	5
F6	11110110	6	6
F7	11110111	7	7
F8	11111000	8	8
F9	11111001	9	9
FA	11111010		
FB	11111011		
FC	11111100		
FD	11111101		
FE	11111110		
FF	11111111		

Appendix C. Powers of Two 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

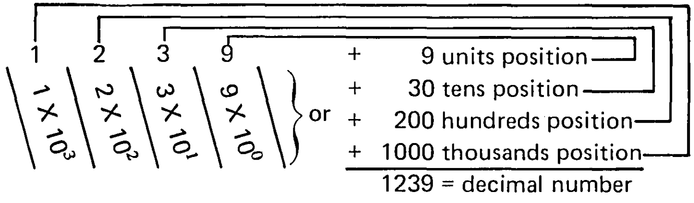
Appendix D. Binary and Hexadecimal Number Notations

BINARY NUMBER NOTATION

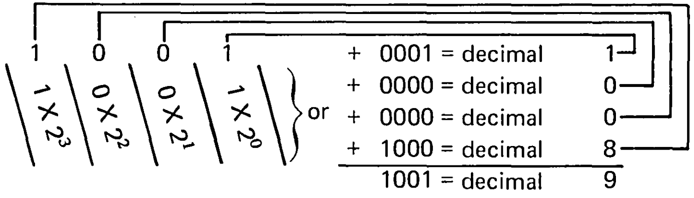
A binary number system uses a base of two. A base-of-two number system can be compared with the base-of-ten (decimal) number system.

Decimal Number	Binary Number Equivalent
0	0
1	1
2	10
3	11
4	100
5	101
6	110
7	111
8	1000
9	1001

Example of a Decimal Number



Example of a Binary Number



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 two in each position. CE panel displays are in binary form: a bit light on is indicated by a 1; a bit light off is indicated by a zero.

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 hex 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 hex position. The following table shows the comparable values of the three number systems:

Decimal	Binary	Hex
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	Hex
16	0001 0000	10
17	0001 0001	11
18	0001 0010	12
19	0001 0011	13
20	0001 0100	14
21	0001 0101	15
etc	etc	etc

Remember that the computer deals only with binary. However, 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 hex value of 1E13. This is easier to state than the string of 1's and 0's.

Appendix E. Hexadecimal-Decimal Conversion Tables

The tables in this appendix provide direct conversion of decimal and hexadecimal numbers in these ranges:

Hex	Decimal
000 to FFF	0000 to 4095

For numbers outside the range of the tables, add the following values to the table figures:

Hex	Decimal
1000	4096
2000	8192
3000	12288
4000	16384
5000	20480
6000	24576
7000	28672
8000	32768

Three-position hex values composed of the numerals listed at the side and top of the tables convert to the decimal values listed inside the tables. Decimal values inside the tables convert to hex values composed of the coordinate numerals at the side and top of the tables.

	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	3819
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 F. Polling and Addressing Characters for Tributary Stations

Polling and addressing characters must be used together in certain pairs; that is, once a polling character is selected, the complementary addressing character is determined; once an addressing character is selected, the complementary polling character is determined.

The pairs of valid polling and addressing characters for both EBCDIC and ASCII are as follows:

EBCDIC

Polling Character	Hexadecimal Representation	Addressing Character	Hexadecimal Representation
BB	C2C2	SS	E2E2
CC	C3C3	TT	E3E3
DD	C4C4	UU	E4E4
EE	C5C5	VV	E5E5
FF	C6C6	WW	E6E6
GG	C7C7	XX	E7E7
HH	C8C8	YY	E8E8
II	C9C9	ZZ	E9E9
JJ	D1D1	11	F1F1
KK	D2D2	22	F2F2
LL	D3D3	33	F3F3
MM	D4D4	44	F4F4
NN	D5D5	55	F5F5
OO	D6D6	66	F6F6
PP	D7D7	77	F7F7
QQ	D8D8	88	F8F8
RR	D9D9	99	F9F9

ASCII

Polling Character	Hexadecimal Representation	Addressing Character	Hexadecimal Representation
AA	4141	aa	6161
BB	4242	bb	6262
CC	4343	cc	6363
DD	4444	dd	6464
EE	4545	ee	6565
FF	4646	ff	6666
GG	4747	gg	6767
HH	4848	hh	6868
II	4949	ii	6969
JJ	4A4A	jj	6A6A
KK	4B4B	kk	6B6B
LL	4C4C	ll	6C6C
MM	4D4D	mm	6D6D
NN	4E4E	nn	6E6E
OO	4F4F	oo	6F6F
PP	5050	pp	7070
QQ	5151	qq	7171
RR	5252	rr	7272
SS	5353	ss	7373
TT	5454	tt	7474
UU	5555	uu	7575
VV	5656	vv	7676
WW	5757	ww	7777
XX	5858	xx	7878
YY	5959	yy	7979
ZZ	5A5A	zz	7A7A

To specify polling or addressing characters in the ADDR-nn parameter of the SETR utility control statement or the OVERRIDE command statement format, give the hex representation of one of the addressing characters. It will be duplicated by the system to provide two characters. At the same time, the corresponding polling characters will be determined.

For example, ADDR-E7 is given to specify the EBCDIC addressing characters XX and the corresponding polling characters GG. ADDR-70 is given to specify the ASCII addressing characters pp and the corresponding polling characters PP.

The ASCII and EBCDIC character sets are shown in the following charts. The BSCA control characters recognized by the system are listed in Chapter 9.

ASCII Codes

		Main Storage Bit Positions 0, 1, 2, 3															
Main Storage Bit Positions		0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
4, 5, 6, 7	Hex	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0000	0	NUL	DLE	SP	0	@	P	`	p								
0001	1	SOH	DC1	!	1	A	Q	a	q								
0010	2	STX	DC2	"	2	B	R	b	r								
0011	3	ETX	DC3	#	3	C	S	c	s								
0100	4	EOT	DC4	$	4	D	T	d	t								
0101	5	ENQ	NAK	%	5	E	U	e	u								
0110	6	ACK	SYN	&	6	F	V	f	v								
0111	7	BEL	ETB	'	7	G	W	g	w								
1000	8	BS	CAN	(	8	H	X	h	x								
1001	9	HT	EM	)	9	I	Y	i	y								
1010	A	LF	SUB	*	:	J	Z	j	z								
1011	B	VT	ESC	+	;	K	[	k	{								
1100	C	FF	FS	,	<	L	\	:	¦								
1101	D	CR	GS	-	=	M	]	m	}								
1110	E	SO	RS	.	>	N	¬	n	~								
1111	F	SI	US	/	?	O	_	o	DEL								

		Main Storage Bit Positions 0, 1, 2, 3															
Main Storage Bit Positions		0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
4, 5, 6, 7	Hex	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0000	0	NUL	DLE	DS		SP	&	-						{	}	\	0
0001	1	SOH	DC1	SOS			•	/		a	j	~		A	j		1
0010	2	STX	DC2	FS	SYN					b	k	s		B	K	S	2
0011	3	ETX	DC3	WUS						c	l	t		C	L	T	3
0100	4	SEL	RES	BYP	PP					d	m	u		D	M	U	4
0101	5	HT	NL	LF	TRN					e	n	v		E	N	V	5
0110	6	RNL	BS	EOB / ETB	NBS					f	o	w		F	O	W	6
0111	7	DEL	POC	PRE / ESC	EOT					g	p	x		G	P	X	7
1000	8	GE	CAN		SBS					h	q	y		H	Q	Y	8
1001	9	RLF	EM		IT				\	i	r	z		I	R	Z	9
1010	A	RPT	UBS	SM	REF	¢	!	¦	:								LVM
1011	B	VT	CU1	FMT	CU3	.	$	,	#								
1100	C	FF	IFS		DC4	<	*	%	@					⑀		⑁	
1101	D	CR	IGS	ENQ	NKA	(	)	_	'								
1110	E	SO	IRS	ACK		+	;	>	=					⑂			
1111	F	S1	IUS	BEL	SUB	|	┐	?	"								EO

⧄ Duplicate Assignment

ONU2318

Glossary

absolute value: The numeric value of a real number irrespective of sign.

address translation registers: Sixty-four 1-byte registers (32 for program-level tasks and 32 for I/O use) that are used to convert the addresses specified by the program into the main storage addresses in which the program actually resides.

alternative: Offering or expressing a choice.

assembler: A computer program that prepares an object program from a source program written in a symbolic source language in which there is a one to one correspondence between the instruction formats and data formats coded and those used by the computer.

batch: A group of jobs to be run on a computer at one time with the same program.

bidirectional printing: The ability of a printer with a print head to print successively left to right and right to left.

buffer: (1) Storage or programming that compensates for a difference in rate of flow or data, or time of occurrence of events, when transmitting data from one part of a computer system to another. (2) An area of storage that is temporarily reserved for use in performing an input/output operation, into which data is read or from which data is written.

control processor: A group of programs that execute control storage instructions that determine channel data transfers and main storage allocation.

element: The smallest addressable unit of an array or table.

error recovery procedure: A set of instructions that helps to isolate and, where possible, to recover from equipment errors. The instructions are often used with programs that record the statistics of machine malfunctions.

execute: To cause a program, utility, instruction, or other machine function to be performed. During execution, information is processed according to machine language instructions to produce the desired output.

initial program load: A sequence of events that loads the system programs and prepares the system for execution of jobs.

integer: A whole number.

invalid character: A character that a machine or program does not recognize.

keylock: A key-operated switch on the display station, which can be used to prevent unauthorized users from operating the system.

keystroke: The act of pressing one key.

logical instructions: Instructions that operate according to the logic of the system; for example, the compare logical immediate instruction.

main storage processor: Hardware that executes system instructions in main storage.

mask: A pattern that controls the keeping, deleting, or testing of portions of another pattern of characters.

megabyte: One million bytes.

menu: A displayed list of items (usually jobs) from which the operator makes a selection.

mutually exclusive: Two or more items, only one of which can be active or true at one time.

null character: A character, hex 00, which blanks the rest of the print line.

object program: A set of instructions in machine language. The object program is produced by a compiler from a source program.

operand: A quantity of data that is operated on, or the address in a computer instruction of data to be operated on.

parameter: (1) A variable that is assigned a particular value for a specific purpose or process. (2) A value specified in a command statement or a control statement.

polling: (1) (SDLC) A technique by which each of the stations sharing a communications line is periodically interrogated to determine whether it requires servicing. (2) (BSC) In a multipoint environment, an invitation to send, transmitted from the primary station to a specific tributary station.

post: To enter a unit of information into a record, or to note the completion of an event.

register: A storage device or circuit that stores those limited elements of data required for the immediate execution of I/O, storage, processing, and control functions, or a group of latches or polarity hold circuits used to store one or two bytes of information.

restore: To bring back or put back into a former or original state.

sector: An area on a disk track or diskette track reserved to record a unit of data.

significant digit: For the insert and test characters instruction, any of the digits 1 through 9.

transient area: An area of main storage or control storage used for temporary storage of transient routines.

transient routine: A routine that is permanently stored on disk and is loaded into the transient area when needed for execution.

twinaxial: A cable made of two twisted wires inside a shield.

Index

Index of Machine Instructions by Machine Code Sequence

READER'S COMMENT FORM

Please use this form only to identify publication errors or request changes to publications. Technical questions about IBM systems, changes in IBM programming support, requests for additional publications, etc, should be directed to your IBM representative or to the IBM branch office nearest your location.

Error in publication (typographical, illustration, and so on). **No reply.**

Page Number *Error*

Inaccurate or misleading information in this publication. Please tell us about it by using this postage-paid form. We will correct or clarify the publication, or tell you why a change is not being made, provided you include your name and address.

Page Number *Comment*

Note: All comments and suggestions become the property of IBM.

○ No postage necessary if mailed in the U.S.A.

Name ______________________________

Address ______________________________

____

IBM System/34
Functions
Reference Manual

SA21-9243-0

SA21-9243-0

Cut Along Line

Fold

Fold

FIRST CLASS
PERMIT NO. 40
ARMONK, N. Y.

BUSINESS REPLY MAIL

NO POSTAGE STAMP NECESSARY IF MAILED IN THE UNITED STATES

POSTAGE WILL BE PAID BY . . .

IBM Corporation
General Systems Division
Development Laboratory
Publications, Dept. 245
Rochester, Minnesota 55901

Fold

Fold

IBM System/34 Functions Reference (File No. S34-01) Printed in U.S.A. SA21-9243-0

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)

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/34 Functions Reference (File No. S34-01) Printed in U.S.A. SA21-9243-0